SHARP

ポケットコンピュータ

形名 **PC-1245**

取扱説明書

保証書付（巻末）
(WITH WARRANTY CARD)

信頼と保証のBMマーク
事務機械工業会 認定

〈はじめに〉

このたびはシャープポケットコンピュータ〈PC-1245〉をお買いあげいただき、まことにありがとうございました。

この計算機は、プログラム言語にパーソナルコンピュータなどで広く使用されていますBASIC(ベーシック)を用いたプログラム計算機です。

BASICは世界でもっとも多くの人が使用しているといわれるプログラム言語で、初めての人でも、少しの学習で容易にプログラムを組むことができます。

本機を正しくお使いいただくために、この取扱説明書をよくお読みください。

なお、この取扱説明書は、「シャープサービス・お客様ご相談窓口一覧表」とともに、必ず保存してください。万一ご使用中にわからないことが生じたとき、きっとお役に立ちます。

(ご注意)

- この取扱説明書の内容は改良のため予告なく変更する場合があります。
- 本機は非常に複雑な機能および組合せを有する製品であり、出荷に際して取扱説明書を含め十分なチェックをして万全を期しておりますが、万一ご使用中ご不審な点お気づきのことがありましたらもよりのシャープサービス・お客様ご相談窓口までご連絡ください。
  なお運用した結果生じる影響については責任を負いかねますので、あらかじめご了承ください。
- 本書に記載のプログラムを使用したことによる金銭上の損害および逸失利益または第三者からのいかなる請求についても当社はその責任を負いませんのであらかじめ、ご了承ください。

目　次

（ページ）

◆おねがい…… 4
●異常が発生した場合の処理について… 5
1．キー操作と表示…… 7
1-1）各部のなまえ…… 7
1-2）スイッチ、キーの操作のしかた…… 8
1-3）モードについて…… 9
1-4）表示の調整…… 10
1-5）表示部横の表の見かた…… 10
2．マニュアル計算…… 11
2-1）基本計算…… 12
2-2）キー操作の訂正…… 14
2-3）プレイバック機能…… 15
2-4）関数計算…… 16
2-5）メモリー計算…… 19
2-6）計算の優先順位…… 21
3．プログラム（その1）…… 23
3-1）プログラムの基本…… 23
3-2）プログラムの入れかたと編集（NEW, LIST）…… 25
3-3）プログラムの実行（RUN）…… 28
4．プログラム（その2）…… 30
4-1）入力命令（INPUT）…… 30
4-2）出力命令（PRINT）…… 32
4-3）代入文（LET）…… 33
4-4）END文 …… 34
4-5）ジャンプ命令（GOTO） …… 34
4-6）判断（IF） …… 35
4-7）くり返し計算（FOR…TO…STEP, NEXT） … 37

（ページ）

4-8）サブルーチン（GOSUB, RETURN）…… 39
5．変数…… 41
5-1）変数の構成…… 41
5-2）固定変数…… 41
5-3）配列変数（DIM） …… 42
5-4）メモリーの構成と変数…… 44
5-5）変数をクリア（消去）するには（CLEAR）…… 46
5-6）変数A（　）について…… 46
6．定義付けプログラム…… 48
7．表示に関するBASIC …… 49
7-1）フォーマット指定（USING）…… 49
7-2）表示時間の指定（WAIT） … 51
7-3）PAUSE命令…… 51
8．そのほかのBASIC …… 53
8-1）READとDATA, RESTORE…… 53
8-2）ON　GOTO, ON　GOSUB…… 54
8-3）REM…… 55
8-4）STOP…… 55
8-5）AREAD…… 56
9．特別な機能…… 57
9-1）乱数関数（RND, RANDOM）…… 57
9-2）INKEY$ …… 58
9-3）音発生機能（BEEP）…… 59
9-4）CONT…… 59

（ページ）

9-5)MEM…… 60
9-6)論理積、論理和、否定
（AND, OR, NOT） …… 60
9-7)条件式の結合…… 62

10. デバッグ
（TR ON, TR OFF）…… 64

11. 文字の編集…… 66
11-1)文字列の分断…… 66
（LEFT$, RIGHT$, MID$）
11-2)文字列の結合…… 67
11-3)文字数を数える方法
（LEN）…… 67
11-4)数値と文字の変換
（VAL, STR$）…… 68
11-5)アスキーコードの変換
（ASC, CHR$） …… 69
11-6)文字列の比較…… 70

12. パスワード(PASS)…… 71

◆プリンタに関する命令…… 72
(1)プリント命令
（LPRINT）…… 72
(2) PRINT→LPRINT指定命令…… 73
(3) マニュアル計算でのプリント …… 74
(4) リストのしかた
（LLIST）…… 74

◆テープレコーダーに関する命令…… 76
(1) プログラムの記録命令
（CSAVE）…… 76
(2) プログラムの転送命令
（CLOAD, MERGE, CHAIN）…… 77
(3) 照合命令
（CLOAD?）…… 81

（ページ）

(4) データの記録命令
（PRINT＃）…… 82
(5) データの転送命令
（INPUT＃） …… 83

◆オプション機器のご案内…… 84

◆計算機とテープレコーダーの
操作方法…… 85

1．CE-125を使用される場合…… 85
(1)準備…… 85
(2)テープへの記録方法…… 85
(3)計算機とテープの内容の照合…… 87
(4)テープからの転送…… 88

2．CE-124を使用される場合…… 89
(1)準備…… 89
(2)テープへの記録方法…… 89
(3)計算機とテープの内容の照合…… 92
(4)テープからの転送…… 93

◆キーの機能…… 95

◆PC-1250／PC-1251用プログラムを
使用する場合の注意点…… 100

◆電源について…… 101

◆仕　様…… 104

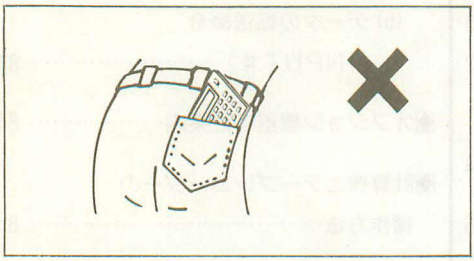
◆アフターサービスについて…… 105

◆ポケットコンピュータ友の会について… 107

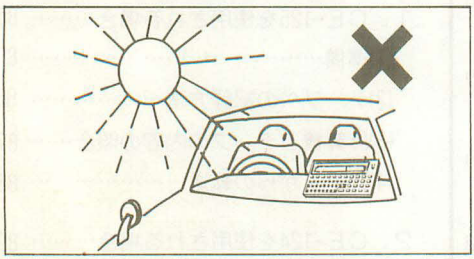
◆CE-125付属のマイクロカセットテープ
御利用に際して…… 108

●エラーコードとエラー内容…… 133
●キャラクタ・コード表…… 135
●関数・命令(省略形)一覧…… 136
●索　引…… 144
保証書…… 147

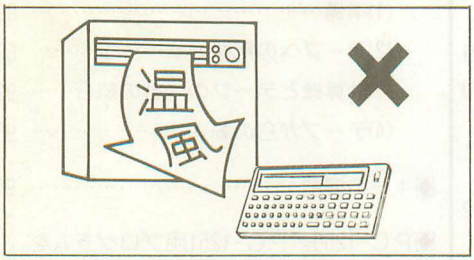
おねがい



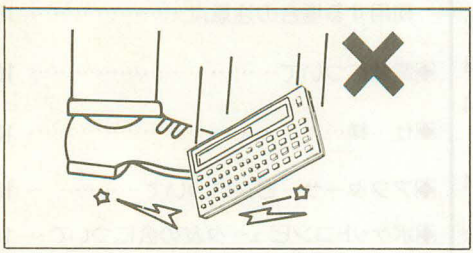
ズボンの後ポケットなどに入れないでください。
座ったときなどに大きな力が加わり、特にガラスでできている液晶表示部が割れることがあります。



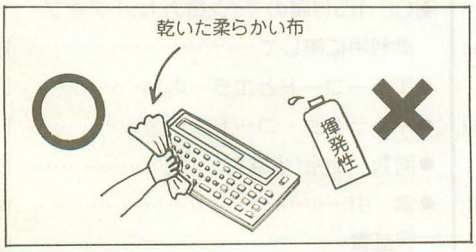
窓をしめきった、日の当たる自動車内に放置したり、直射日光の当たる場所に置いたりしないでください。
高温により故障の原因になります。



暖ぼう器具の近くなど高温になる場所に置かないでください。



落としたり、ぶっつけたりしないでください。



お手入れに揮発性の液体(シンナー、ベンジン)など)や、ぬれぞうきんなどは使用しないで、乾いた柔らかい布をご使用ください。

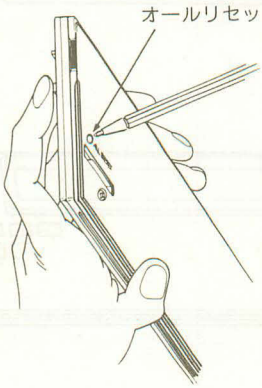
異常が発生した場合の処理について

本機をご使用中に強度の外来ノイズや強いショックをうけた場合などに、ごくまれに ON BRK キーや電源／モードスイッチも含めた、すべてのキーの機能が働かなくなるなどの異常が発生することがあります。

本機にこのような異常が発生した場合は、次の操作のうち、どちらかをおこなってください。

① **任意のキーを押したまま計算機裏面のオールリセットスイッチを押します。**

この場合は、プログラム、データなどは保持されます。

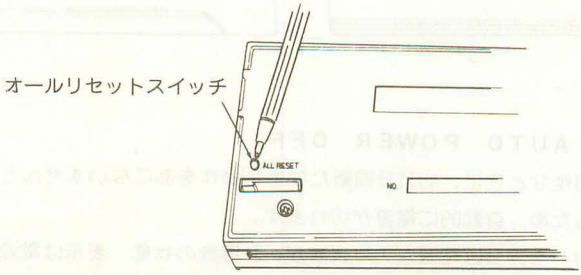


（オールリセットスイッチを操作するときはボールペンなどで押してください。しん先の出たシャープペンシルなど、先の折れやすいもの、また、とくに針などの先のとがったものは使用しないでください。）

② **キーを押さずに、オールリセットスイッチのみを押します。**

この場合は、プログラム、データなどはすべて消去されます。

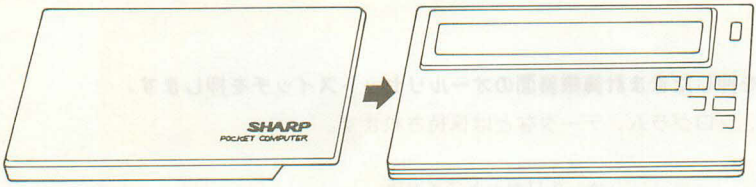
①の方法で異常状態を解除した後、再び異常が発生する場合は、②の方法で解除し、もう一度プログラムを入力し直してください。



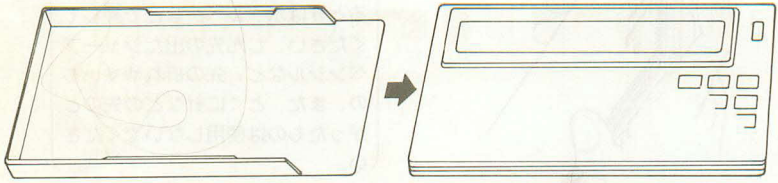
＝ハードカバーの使いかた＝

計算機を使わないときは、ハードカバーを計算機の表側に取りつけてください。
計算機を使用するときは、計算機の裏側に取りつけてください。

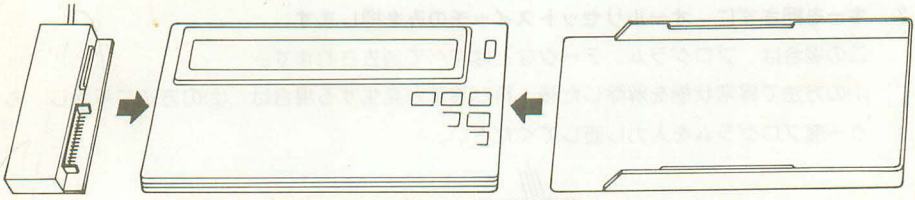
●計算機を使わないとき。



●計算機を使うとき。



●カセットインターフェイス（CE-124：別売）を使うとき。

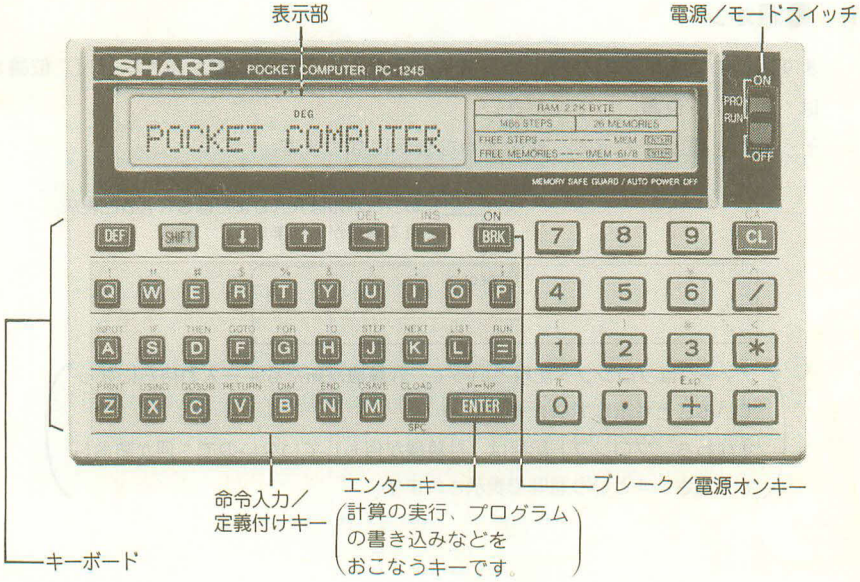


●**オートパワーオフ（AUTO　POWER　OFF）**

本機はキー入力が可能なときに、約11分間新たなキー操作をおこないませんと、電池の消耗を少なくするため、自動的に電源が切れます。
この場合、ON BRK キーを押せば電源が入りますが、計算機の状態、表示は電源が切れる前の状態にもどります。ただし、電源／モードスイッチを別の位置にしていた場合は、前の状態にもどりません。

1．キー操作と表示

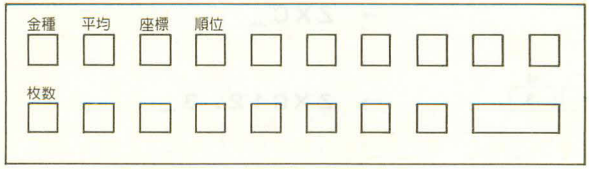
1-1）各部のなまえ



テンプレート

テンプレートは定義付けキーに定義付けしたプログラムの名称などを書き込み、キーボードにはめ込んで使用するものです。

例　定義付けしたプログラムの名称などを書き込んで使用（定義付けプログラム：48ページ参照）



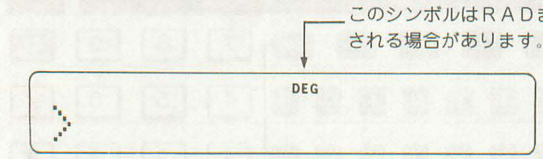
●テンプレートをお求めの際は、本機お買いあげ販売店または別紙一覧表に記載のお客様ご相談窓口（技術センター）に「PC-1245用テンプレート」と指定してお申しつけください。

1-2）スイッチ・キーの操作のしかた

本機は、上面にたくさんの(52個の)キーと、ひとつのスイッチがあります。
ここでは、これらの操作のしかたを説明します。

(1) 電源オン

まずはじめに表示部の右側にあるスイッチを“ＯＦＦ”位置から“ＲＵＮ”位置または“ＰＲＯ”位置にしてください。
そうすると電源が入って、次のような表示になります。



（ ＞マークはプロンプト記号といって、計算機が新たなキー入力待ちになっていることを示しています。
すなわち、プロンプト記号は、計算機が何もしていないので“何か命令してください”という意味で表示されます。 ）

(2) キーの操作のしかた

それでは、まずキーを押してみましょう。スイッチを“ＲＵＮ”位置にして、次のようにキーを押し、表示を見てください。

（例）

PRINT USING GOSUB
Z X C → ＺＸＣ_

( ) √ @
1 2 • 3 → ＺＸＣ１２．３_

CA
CL → ＞

INPUT RUN EXP
A = 4 + 5 → Ａ＝４＋５_

CA
CL は表示をクリア(消去)するときや、エラーを解除するときに押します。

これはカーソルといいます。
次になにか入力すると、カーソルがある場所に入ります。

このように、アルファベットのキーや数字キーなどを押せば、それぞれのキーに直接記されている文字や数字、記号が入力されます。

キーの上側に茶色で記されている文字や記号を入力したいときは、それぞれのキーの前に [SHIFT] キーを押します。

（例）

表示をクリアします。

[CA] [CL] [SHIFT] [PRINT] [Z] → PRINT _

[SHIFT] [" W] [SHIFT] [(1] [SHIFT] [√ •] → PRINT " (√_

このように [SHIFT] キーは、2つの機能を持つキーの茶色で記されている機能を使用するときにつかうキーです。

[SHIFT] キーを何回か押してみると、表示部の右上で SHIFT シンボルがついたり消えたりします。

SHIFT シンボルは [SHIFT] キーが押され、茶色で記されている機能が指定されていることを示すシンボルですので、このシンボルを確認しながら、キーを操作してください。

- [CLOAD / SPC] キーは、そのまま押せばスペース（空き）が入力されます。
- [ON / BRK] は、オートパワーオフ機能（6 ページ参照）により電源が切れているときには電源オンキーとして働き、プログラム実行中に押せば、ブレーク（一時停止）キーとして働きます。

1-3）モードについて

この計算機では、プログラムを記憶させるときと、プログラムを実行するときでは、ちがうモード（状態）でおこないます。

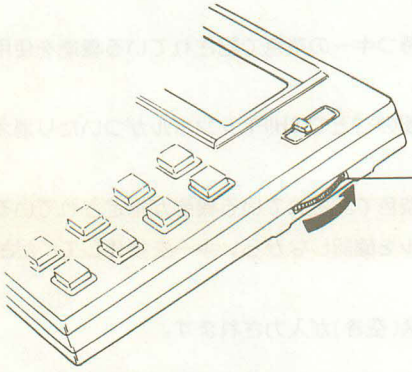
モードは、電源／モードスイッチで、次のように指定することができます。

PRO位置： PRO（プログラム）モードが指定されます。（PROGRAMモード）
プログラムを計算機に記憶させたり、プログラムの修正、訂正をおこなうときは、スイッチをこの位置にします。

RUN位置： RUN（ラン）モードが指定されます。（RUNモード）
プログラムを実行させたり、電卓のようにマニュアル計算をするときは、スイッチをこの位置にします。

プログラム実行中に電源／モードスイッチをＯＦＦ位置にしますと、そのとき実行していたラインを終了してから、あるいはそのラインを抜けたとき電源が切れます。
したがって、実行時間が非常に長い命令がそのラインにありますと、なかなか電源が切れません。
このようなときは ON BRK キーにより、命令の実行を止めてからスイッチをＯＦＦにしてください。

1-4）表示の調整



コントラストボリューム

矢印の方向に回せば表示が濃くなり、逆方向に回せば薄くなります。
表示が見やすくなるように調整してください。

1-5）表示部横の表の見かた

本機で使用できるプログラムの大きさを示しています。
（1486ステップあるいは1486バイト）

本機内のＲＡＭ容量を示しています。
（ＲＡＭ2.2キロバイト）

メモリー（固定変数）の数を示しています。
（26メモリー）

現在、空いているプログラム・データエリアの大きさを求める方法を示しています。
（60ページ参照）

1つの変数名（配列変数）で使える変数の数を求める方法を示しています。
（45ページ参照）

RAM 2.2K BYTE	
1486 STEPS	26 MEMORIES
FREE STEPS ..... MEM ENTER	
FREE MEMORIES .....(MEM-6)/ 8 ENTER	

RAM ：Random Access Memory
ラムと言い、読み書きができるメモリーです。
書き込みができない、読み出し専用のメモリーをROM(ロム：Read Only Memory)と言います。

ステップ：プログラムを構成する１つの命令、１つの記号、文字などを数えるときの単位のことで、プログラムの大きさを表わすときなどに使います。

バイト ：計算機内で扱う情報単位のことで、メモリーの大きさ(容量)やプログラムの大きさを表わす場合などに使用します。

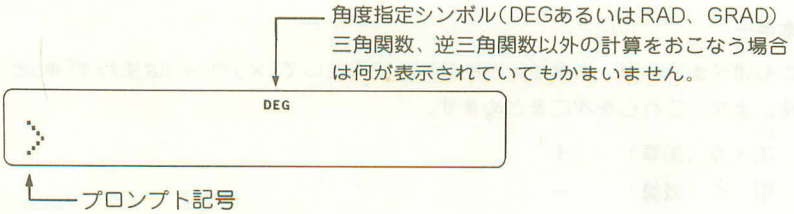
●ステップとバイトはプログラムの大きさを表わす場合は同じ意味（１ステップ＝１バイト）になりますので、本書では特に必要と思われる場合以外は、すべてバイトで表わします。

２．マニュアル計算

マニュアル計算とは、電卓と同じように１つ１つキーを押して計算をおこなっていく方法です。
この章では、本機の操作に慣れ親しんでいただくために、基本的な計算をマニュアルでおこなう方法について説明します。

準備：電源オンとモード指定

マニュアル計算は、ＲＵＮモードでおこないますので、まず電源／モードスイッチをＯＦＦ位置からＲＵＮ位置にしてください。
電源が入り、表示部の左端に ＞(プロンプト記号)が表示されます。



＝ご注意＝

本機は、数字のゼロ(0)を∅と表示します。
これは、アルファベットのオー(O)と区別するためで、本書においても、オーとゼロの区別がつきにくい場合はゼロを∅と表わしています。

2-1）基本計算

それでは計算をおこないますが、計算をおこなうとき、ちょっと注意していただきたいことがあります。通常の数学では四則計算の記号として「＋，－，×，÷」を使いますが、BASICでは「×」や「÷」は使わず、かわりに「＊」と「／」を用いて計算をおこないます。また、計算結果は [=] ではなく [ENTER] キーで求めます。
この点に注意して計算をしてください。

（例）

① 2＋3×4＝

[2] [+] [3] [*] [4] [ENTER]　　14.

② 5×(－6)＋7＝

[5] [*] [-] [6] [+] [7] [ENTER]　　－23.

③ (6＋5)÷(4－1)＝

[SHIFT] [(] [1] [6] [+] [5] [SHIFT] [)] [2] [/]

[SHIFT] [(] [1] [4] [-] [1] [SHIFT] [)] [2] [ENTER]　　3.666666667

④ $-5\times10^{3}\div(4\times10^{-3})=$

[CL] [-] [5] [SHIFT] [EXP] [+] [3] [/] [4] [SHIFT] [EXP] [+] [-] [3] [ENTER]

－1250000.

⑤ 500000×43200＝

[5] [0] [0] [0] [0] [0] [*] [4] [3] [2] [0] [0] [ENTER]

2.16E 10

◉ **基本キー**

先にも述べましたが、本機では四則計算の記号として「×」や「÷」は使わず「＊」と「／」を使います。これらを次にまとめます。

加える（加算）	＋
引　く（減算）	－
かける（乗算）	＊（アスタリスク）
割　る（除算）	／（スラッシュ）
符　号	－または＋
指　数	E（Exp）
計算の実行	[ENTER]

◉ 数式の長さ

計算をおこなうときは、数式の通りにキー入力し、最後に [ENTER] キーで計算を実行させますが、1回の計算で入力することができる数式の長さ([ENTER] キーを押す前までの長さ)は79キー操作までです。

80キー操作目を入れますと、その内容と“▮”が交互に表示されます。(80キー操作目の内容は無効になります。)

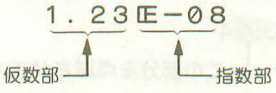
◉ 計算結果の表示方法と計算桁数

数値を表わす方法として、通常よく用いられるのは、たとえば120000や0.00123のような方法ですが、大きな数値や小さな数値では、たとえば2.5×10^{12}や1.23×10^{-8}のような形で表わすことがあります。

数値をこのような形で表わす方法を**浮動小数点方式(指数方式)**と呼びます。

そして、2.5×10^{12}の“2.5”や1.23×10^{-8}の“1.23”を**仮数部**と呼び、“10^{12}”や、“10^{-8}”を**指数部**と呼びます。(数値を120000や0.00123のように表わす方法を**固定小数点方式**と呼びます。)

本機では、数値を浮動小数点方式で表示する場合は、指数部を示す記号として **E** を入れて**2.5E12**や**1.23E−08**のように表示します。



また、キー入力する場合は、 [2] [•] [5] [SHIFT] [EXP +] [1] [2] や [1] [•] [2] [3] [SHIFT] [EXP +] [−] [8] のようにキーを押します。

なお、指数部に3桁以上の数値を入れた場合は、後2桁の数字が有効になります。

(例) [5] [SHIFT] [EXP +] [1] [2] [3] [ENTER] → 5．E 23

◉ 連続計算のしかた

[ENTER] キーを押しますとその計算結果が表示されますが、その結果を続けて計算することもできます。

(例) [1] [+] [2] [ENTER] 3．

[*] [4] [ENTER] 12．(3×4を実行)

◉ エラー表示

マニュアル計算でエラーになりますと

ERROR 1 や ERROR 2

のようにエラー番号が表示されます。(エラー番号の意味は133ページの「エラーコードとエラー内容」を参照ください。)

＝注＝

本書では、数字やアルファベットの入力、数式の入力などのキー操作においては、キーのワクや、SHIFT キーの操作などを省略しています。

（例）

2 ＊ 3 ・ 4 SHIFT ＋(Exp) － 8 → ２＊３．４Ｅ－８と記載

SHIFT 1(() 5 ＋ 6 SHIFT 2()) ／ 4 → （５＋６）／４と記載

2-2）キー操作の訂正

もしも誤ってキー操作をしてもあわてずに次の方法で訂正しましょう。

⑴ **誤りが多いとき**

CL キーを押せば表示が消えて、左端に〉記号が表示されますので、最初から入れ直します。

⑵ **誤って他のキーを押したとき**

誤って押したキーのシンボルが、点滅表示されるまで ◄(DEL) キーを押して、そこで正しいキーを押します。（ ►(INS) や ◄(DEL) は押し続けるとカーソル※が続けて移動します。）

（例）　**２＋３＊４を２＋３／４とした場合は**

２＋３／４

└この部分を点滅させて＊を押し、ENTER キーを押します。

※カーソル

キー操作をおこなうとき＿を表示したり、キー操作の訂正のときに点滅表示をしたりしますが、これらは次に操作をしたキーがこの部分に入ることを示しており、これをカーソルと呼びます。

⑶ **誤って余分なキーを押したとき**

余分なキーのシンボルが点滅表示されるまで ◄(DEL) キーを押し、SHIFT ◄(DEL) と押します。（DEL：DELETE（デリート）…削除を意味します。）

（例）　**２＋３＊４を２＋３＊／４とした場合は**

◄(DEL) **を操作**　　２＋３＊／４
└この部分を点滅させてください。

SHIFT ◄(DEL) **と押す**　　２＋３＊４
└ここが点滅します。

ENTER **を押す**　　１４．（答）

(4) キーの操作がぬけたときは

ぬけたキー操作部分の次のキーシンボルが点滅するまで [DEL ◄] キーを押し,そこで [SHIFT] [INS ►] と押すと一文字分の空欄ができますので、次にぬけていたキーを押せば、その位置に入力されます。(INS：INSERT(インサート)…挿入を意味します。)

(例) 2+3*4 を 2+34 とした場合は

[DEL ◄] を操作　　2+3[4]
　　└この部分を点滅させてください。

[SHIFT] [INS ►] と押す　　2+3[4]
　　└ここが点滅します。

[*] を押す　　2+3*[4]
　　└ここが点滅します。

[ENTER] を押す　　14.　(答)

2-3) プレイバック機能

[ENTER] キーで計算を実行したのち、[DEL ◄] あるいは [INS ►] キーを押すと実行した式が呼びもどされます。これは実行した式の確認や実行した式の一部を変更して再度実行したいときに便利です。特に [ENTER] で計算を実行した結果、エラーになったときにこのプレイバックを使いますと、エラーを起こした場所をカーソルで示していますので、エラーの原因がわかりやすく大変便利です。(>記号(プロンプト記号)が表示されたときには、すでに内容が消されていますのでプレイバックできません。)

(例1) 36／1+2) [ENTER] と押してエラーになったとき

[DEL ◄] を押すと　　36／1+2[ ]
　　└点滅表示(カーソル)

(例2) 36／(1+2) [ENTER] と押したとき

36／(1+2) [ENTER]　　12.

[INS ►]　　[3]6／(1+2)
　　└点滅表示(カーソル)

[ENTER]　　12.

[DEL ◄]　　36／(1+2)_
　　└カーソル

2-4）関数計算

本機には、次の基本関数があります。

項　　目	一般の表示	本機のキー操作	備　　考
三角関数	sin cos tan	SIN COS TAN	
逆三角関数	$\sin^{-1}$ $\cos^{-1}$ $\tan^{-1}$	ASN ACS ATN	
常用対数	log	LOG	$\log_{10} x$（10を底とする対数）
自然対数	ln	LN	$\log_e x$（eを底とする対数）
指数関数	e^x	EXP	e≒2.718281828
べき乗		^	AへBにおいて A^B
開平	$\sqrt{\ }$	$\sqrt{\ }$あるいはSQR	
10進数(度)→60進数(度分秒)変換		DMS	角度変換（DEG0.5をDEG.5のように0を省略しないでください。）
60進数(度分秒)→10進数(度)変換		DEG	
整数化（インテジャー）		INT	INT(x)において、x以下でかつもっとも大きい整数を求めます。
絶対値（アブソリュート）	|X|	ABS	ABS(x)において、xの絶対値を求めます。
符号関数（シグナム）		SGN	SGN(x)において、$x>0$のとき1、$x<0$のとき−1、$x=0$のとき0が得られます。
円周率	π	πあるいはPI	π≒3.141592654

◉ **角度指定**

三角関数、逆三角関数では扱うことのできる角度単位が3種類ありますので、計算する角度の単位を正しく指定しておく必要があります。

角度単位	表示シンボル	指定のしかた	備　　考
度	DEG	DEGREE ENTER	直角を90で表わす単位〔°〕
ラディアン	RAD	RADIAN ENTER	直角を$\frac{\pi}{2}$で表わす単位〔rad〕
グラード	GRAD	GRAD ENTER	直角を100で表わす単位〔g〕

（$90° = \frac{\pi}{2}$〔rad〕$= 100^g$）

●指定された角度単位はシンボルとして表示部に表示されます。

（DEG、RAD、GRAD）

（計算例）

① $\sin 30° + \cos 60° \times \tan 45° =$

DEGREE [ENTER]　（角度単位を“度”に指定）

SIN30＋COS60＊TAN45 [ENTER]　　1.

② $\tan \frac{\pi}{4} =$

RADIAN [ENTER]　（角度単位を“ラディアン”に指定）

TAN(π／4) [ENTER]　　1.

③ $\cos^{-1}(-0.5) =$

DEGREE [ENTER]

ACS−0.5 [ENTER]　　120.
（120°）

④ $\sqrt{3^2+4^2} =$

√(3^2+4^2) [ENTER]　　5.

⑤ $\log 5 + \ln 5 =$

LOG5＋LN5 [ENTER]　　2.308407917

⑥ $e^{2+3} =$

EXP(2+3) [ENTER]　　148.4131591

⑦ 60進数の30度30分は10進数の何度？

DEG30.30 [ENTER]　　30.5
（30.5度）

⑧ 10進数の30.5度は60進数の何度何分何秒？

DMS30.5 [ENTER]　　30.3
（30度30分）

〈関数の計算範囲〉

関　　数	計　算　範　囲
y へ x（y の x 乗）	$-1\times10^{100}<x\log y<100$ （$y=0$、$x\leqq0$ のときはERROR 2／$y=0$、$x>0$ のときは0／$y<0$、$x\neq$整数のときはERROR 2）下記の例を参照
SIN x COS x TAN x	DEG：$\lvert x\rvert<1\times10^{10}$ RAD：$\lvert x\rvert<\frac{\pi}{180}\times10^{10}$ GRAD：$\lvert x\rvert<\frac{10}{9}\times10^{10}$ ただし、TAN x において次の場合は除く DEG：$\lvert x\rvert=90(2n-1)$ RAD：$\lvert x\rvert=\frac{\pi}{2}(2n-1)$ GRAD：$\lvert x\rvert=100\ (2n-1)$（$n$は整数）
ASN x ACS x	$-1\leqq x\leqq1$
ATN x	$\lvert x\rvert<1\times10^{100}$
LN x LOG x	$1\times10^{-99}\leqq x<1\times10^{100}$
EXP x	$-1\times10^{100}<x\leqq230.2585092$
$\sqrt{x}$	$0\leqq x<1\times10^{100}$

上記以外の関数は、演算数 x が次の範囲で計算可能です。

$1\times10^{-99}\leqq\lvert x\rvert<1\times10^{100}$および0

（例）　0 ^ 0 [ENTER] → ERROR 2
　　0 ^ 5 [ENTER] → 0.
（−4）^ 0．5 [ENTER] → ERROR 2

●関数の精度は原則として10桁目に±1（指数表示の場合は仮数部の最下位桁に±1）の誤差となります。

注）べき乗計算は $\log x$ と 10^x 計算により、計算がおこなわれています。

Y ^ X → $10^{X\log Y}$により計算

したがって、計算機内部では真値との間にある程度の差が生じることは避けられません。通常、この差はほとんど表示上には現われませんが、連続計算をおこなうなど、計算の内容によっては、差が累積されて、表示上に現われる場合もあります。

（例）16−2 ^ 2 ^ 2 [ENTER]　　6．E−10

＝参　考＝

論理演算子（＝、＞、＜、＞＝、＜＝、＜＞）を用いて構成された式（論理式）は、それぞれ次の値をとります。

x 、 y は数値を示します。

＝※	$x=y$ のとき：1 $x \neq y$ のとき：0	＞＝	$x \geqq y$ のとき：1 $x < y$ のとき：0
＞	$x > y$ のとき：1 $x \leqq y$ のとき：0	＜＝	$x \leqq y$ のとき：1 $x > y$ のとき：0
＜	$x < y$ のとき：1 $x \geqq y$ のとき：0	＜＞	$x \neq y$ のとき：1 $x=y$ のとき：0（＜＞は≠と同意味）

※＝を用いる論理式を例えばＡ＝数値、Ｂ＝式のような形にしますと、計算機は論理式と判断せず、変数への入力(代入文)と判断しますので、このような場合は数値＝Ａ、式＝Ｂのような形にしてください。ただし、ＩＦ文における条件式の場合はこの限りではありません。

2-5）メモリー計算

メモリーについてのくわしい説明は41ページでおこないますが、ここでは数値や文字を入れておくデータメモリーについて簡単に説明します。

データメモリーは、その中に数値や文字を入れておき必要に応じて、その内容を呼び出して使用したり、また、内容を入れ替えたりします。

たとえば　Ａ　というデータメモリーを使って、　Ａ＋３　という式を作ったとしますと、同じ式でもメモリーＡの内容を入れ替えれば答は変ってきます。

このように式の中などで、データメモリーの内容を変化させることができることから、データメモリーのことを**変数**と呼びます。

したがって、本書でもデータメモリーを変数と呼びます。

変数は、次のように分類されています。

変数(データメモリー)
- **数値変数(数値メモリー)**：10、500、5正12など、数値を入れます。
- **文字変数(文字メモリー)**：ABC、BASICなど、文字列を入れます。

◉ **変数の名前**

変数の名前はＡ、Ｂ、Ｃ…のように、アルファベット１文字を使用してください。

そして、文字を扱う場合（文字変数）には、アルファベットの後に＄マークをつけて、Ａ＄、Ｂ＄、Ｃ＄…のようにします。

◉ 数値変数への数値の格納（代入）

変数に数値を格納する場合は次の形で指定します。

数値変数＝式　　（式とは、数値、数式、数値変数を指します。）

（例）　A＝3 [ENTER]　……変数Aに3が入ります。

A＝A＋1 [ENTER]　……変数Aの内容に1を加えた結果が変数Aに入ります。

（先にAに3が入っていた場合は3＋1の結果4がAに入ります。）

●数値変数には仮数部10桁、指数部－99～99までの数値を格納することができます。

（変数に数値や文字などを入れる（格納する）ことを“代入”と呼びます。また、A＝3やX$＝"ABC1234"のように数値や文字を代入するための式を代入式と呼びます。）

◉ 文字変数への文字の格納（代入）

変数に文字を格納する場合は次の形で指定します。

文字変数＝"文字"

文字変数＝文字変数

（変数名の後に$（ドルマーク）をつけることにより文字変数を表わします。

（例）A$＝"SHARP" [ENTER]　……A$にSHARPが入ります。

B$＝"HAKATA" [ENTER]　……B$にHAKATAが入ります。

C$＝A$ [ENTER]　……C$にA$の内容、すなわちSHARPが入ります。

●文字変数には最大7文字まで格納することができます。

（注）数値変数A～Zも文字変数A$～Z$も同じメモリーが使用されます。

例えば、数値変数Aと文字変数A$は同じAというメモリーを使用することになります。したがって変数Aと変数A$は同時に存在できません。

（例）　A＝123 [ENTER]　→変数Aに123を格納

A$＝"ABC" [ENTER]　→変数Aを消去し、A$にABCを格納

A$ [ENTER]　→ABC表示

A [ENTER]　→エラー

（計算機は文字変数が数値変数の形で指定されたものとしてエラー表示します。（ERROR 9））

（計算例）

(1) ㋑ 変数Aに3、変数Bに4を入れてください。

A＝3 ENTER
B＝4 ENTER
} または A＝3，B＝4 ENTER

㋺ $\sqrt{A^2+B^2}$を計算して、その結果を変数Cに入れてください。

C＝√(A＊A＋B＊B) ENTER

㋩ 変数A、B、Cの内容を確認してください。

A ENTER　3.
B ENTER　4.
C ENTER　5.

(2) ㋑ 変数X$に"ABC1234"、Y$に"BASIC"という文字を入れてください。

X$="ABC1234" ENTER
Y$="BASIC" ENTER

㋺ 変数X$、Y$の内容を確認してください。

X$ ENTER　ABC1234
Y$ ENTER　BASIC

2-6）計算の優先順位

本機ではカッコ、関数も含めて数式通りのキー操作で計算をおこなうことができます。計算の優先順位の判断や途中結果の処理はすべて計算機が自動的に処理してくれます。計算の優先順位は次の通りです。

1．πや変数の呼び出し
2．2A^3のように、変数やπの直前の＊を省略した乗算が直前にある場合のべき乗
（2A^3=2＊(A^3)）
3．2A、πB、ABのように＊を省略した乗算
4．関数（sin　cosなど）
5．べき乗（^）（2項以外のべき乗）
6．符号（＋，－）
7．乗除算（＊，／）
8．加減算（＋，－）
9．大小比較（>，>=，<，<=，<>）
10．AND，OR

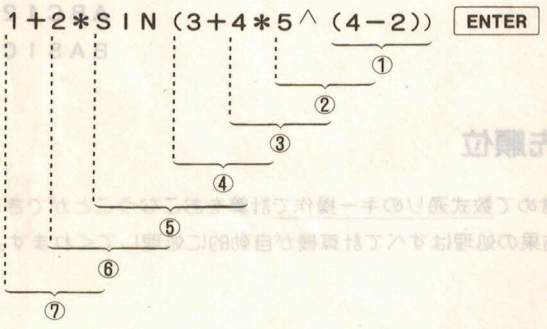
注）●カッコが使用されている場合はカッコ内の計算が最優先されます。
●複合関数（sin cos^{-1} 0.6など）は右から左の順で計算されます。
●べき乗の連算（3^{4^2} 即ち3^4^2など）は右から左の順で計算されます。
●5．と6．では後に出てきたほうが優先順位が高くなります。

（例） −2へ4 → −(2^4)
3へ−2 → 3^{-2}

＝参　考＝

本機では数式通りの計算をおこないますが、数式通りの計算を進めるためには、すぐに処理できない命令やデータ(数値)を一時記憶しておく場所が必要になります。
この一時記憶の場所をバッファといい、本機ではファンクション用バッファを16段、データ用バッファを8段持っています。

（例）1＋2＊SIN（3＋4＊5^（4−2））ENTER と押したときの計算順序と、バッファの使用状況



データバッファ							ファンクションバッファ									
1段	2段	3段	4段	5段	6段	……	1段	2段	3段	4段	5段	6段	7段	8段	9段	……
4	5	4	3	2	1		−	(	^	＊	＋	(	SIN	＊	＋	

例では①→⑦の順で計算が実行されますが、①の計算をおこなうために、②以降の計算を上図のように保存します。
そして、1段目から、計算用のメモリーに呼び出して計算を進めます。
上例のようにカッコもファンクションバッファに記憶されます。
なお、カッコは最高15段までなら、ファンクションバッファを越えない範囲で深みを持たせることができます。

なお、本機では、例えば、2＊Aや3＊πなどのように、メモリーやπの直前にくる乗算命令＊は、省略して2A、3πのような形で入力することができます。
この形での乗算は、関数よりも優先して実行されます。また、この乗算がべき乗の直前にある場合はべき乗が優先されます。

例　SIN2A→SIN(2＊A)と等価
　　2πA＾3→2＊π＊(A＾3)と等価

ただし、この＊を省略した形で入力した〈式〉も、実行時には＊命令があるものとしてファンクションバッファが使用されます。

3．プログラム（その1）

プログラムとは、計算などをおこなうために計算機にその手順を指令する命令書のようなものです。
この命令を計算機が理解できるように書き表わしたり、計算機に覚えさせることをプログラミングとか、プログラムを組むなどと言います。
ここでは、プログラミングの基本について説明します。

3-1）プログラムの基本

ここでは、プログラムの基本的な形について説明します。
まず、次のプログラムを見てください。

〈プログラム1〉

```
10  INPUT  R

20  S=π*R*R
30  PRINT  S

40  END
```

このプログラムは、半径を入力して円の面積を求めるプログラムです。
BASICをまったく知らない人でもこのプログラムが計算機に何を伝えているのか想像がつくと思います。これは次のような指示をするプログラムなのです。

```
10  INPUT  R   ←データを読み込み、Rに入れなさい。
20  S=π*R*R   ←πにRの値を2回掛けなさい。そして、その結果をSに入
                れなさい。
30  PRINT  S   ←Sの値を表示しなさい。
40  END        ←これでプログラムは終りです。
```

プログラムの大部分は、このように、

①データを読み込み、

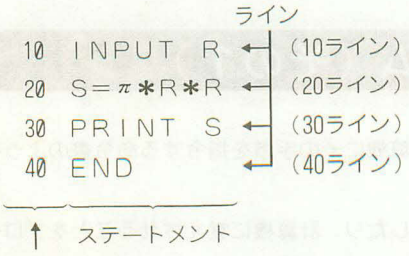
②データを処理して、

③結果を出力する。

という形になっています。

どんなに長いプログラムであっても、基本的には上記の①～③のくり返しあるいは①、②、③それぞれを何回もくり返す、という形になっています。したがって、プログラムはこの形が基本になります。

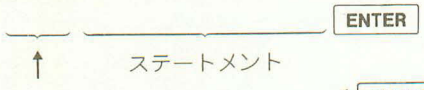
さて、ここで**プログラムの構成**について少しくわしく説明します。



◉ **ライン**

BASICでは、プログラムはいくつかのラインから構成されています。そして1つのラインは、1つのラインナンバーと、いくつかの意味を持ったステートメント（命令など）からなります。

（各ラインの終りは ENTER キーで指定します。）



（ ENTER はラインの終りを意味し、計算機にプログラムを入れるときには必ず必要です。）

● 1つのラインの長さは ENTER キーも含めて最大80キー命令（あるいは最大80バイト）までの長さにすることができます。

◉ **ラインナンバー**

ラインナンバーは1～999の整数をつけます。

プログラムの実行は若いラインナンバーから順次実行されますので、計算の順に若いラインナンバーをつけます。ただし、あとからプログラムの途中にラインを追加したい場合も出てきますので、そういったことを配慮して10、20、30……のように10番おきにラインナンバーをつけるのが最も一般的です。

◉ **ステートメント**

1つの意味を持った処理式や命令をステートメントといいます。

1つのラインは1つ以上のステートメントからなり、2つ以上のステートメントになる場合はステートメントとステートメントの間に：(コロン)を入れて区別します。

＿＿＿＿　＿＿＿＿：＿＿＿＿：＿＿＿＿ ENTER

ラインナンバー　ステートメント　ステートメント　ステートメント

3-2）プログラムの入れかたと編集(NEW、LIST)

◉ **プログラムの入れかた（書き込み）**

23ページのプログラム（プログラム1）を計算機に書き込み、覚えさせてみましょう。

プログラムを計算機に書き込むときは、次の手順でおこないます。

① まず、電源／モードスイッチを"PRO"位置にしてください。(PROモードにしてください。)

② 次に NEW ENTER と押してください。
これで、先に記憶されている（かもしれない）プログラムやデータをすべて消します。

③ 続いてプログラムを1ラインずつ順番に入れてください。

たとえば、10ラインを入れるときは、次のように押します。

1 0 SHIFT A(INPUT) R ENTER　　→10：INPUT R

(または 1 0 I N P U T R ENTER でも同じ)

このとき、注意していただきたいことは、1ライン入れ終るごとに ENTER キーを押すことです。

ENTER キーは、ラインの終りに“このラインは終りました”という意味で必らず必要ですので、忘れないで、プログラムを入れてください。

PROモード

NEW ENTER
10 INPUT R ENTER
20 S＝π＊R＊R ENTER
30 PRINT S ENTER
40 END ENTER

INPUTは SHIFT A(INPUT) または I N P U T と押します。

PRINTは SHIFT Z(PRINT) または P R I N T と押します。

ENDは SHIFT N(END) または E N D と押します。

プログラムを入れていただければ、おわかりと思いますが、各ラインを入れ、[ENTER]を押しますと、ラインナンバーの後に自動的に：(コロン)が入ります。また、ＩＮＰＵＴやＰＲＩＮＴを１文字ずつ入れたときは、これらの命令の後はあいていませんが、[ENTER]キーを押すと、自動的に１桁あけられます。(これらの命令がラインの先頭ではなく、途中にある場合はその前および後の命令や変数、記号などの間が１桁分あけられます。これはＳＩＮ、ＣＯＳなどの関数も同様です。)

このことにより、計算機が、最初の数字をラインナンバー、アルファベットを１つの命令（ＩＮＰＵＴという命令)として受け取り(理解し)、覚えたことを示します。

◉ **プログラムの編集**

プログラムを作成する場合、一度で正しくは、なかなか書けないものです。キー操作のミスや勘違いでまちがえることもあります。このような場合の処理方法について次に説明します。

○表示されているライン内の訂正や追加、削除は14ページのマニュアル計算での説明と同じように [DEL ◄] または [INS ►] キーでカーソルを移動させ、[SHIFT] [DEL ◄] や [SHIFT] [INS ►] を使っておこないます。

（例）　20　Ｓ＝π＊Ｒ＊Ｒ　を
20　π＊Ｒ＊Ｒ＝Ｓ [ENTER] と押した場合の訂正

[INS ►] または [DEL ◄] を操作　　２０　π＊Ｒ＊Ｒ＝Ｓ
ここを点滅させます。

[SHIFT] [INS ►] を操作　　２０　□□π＊Ｒ＊Ｒ＝Ｓ
必要な桁数分をあけます。

[Ｓ] [＝]　　２０　Ｓ＝π＊Ｒ＊Ｒ＝Ｓ
追加します。

[INS ►] を操作　　２０　Ｓ＝π＊Ｒ＊Ｒ＝Ｓ
ここを点滅させます。

[SHIFT] [DEL ◄] [SHIFT] [DEL ◄]　　２０　Ｓ＝π＊Ｒ＊Ｒ＿
不要な内容を削除します。

[ENTER]　　２０：Ｓ＝π＊Ｒ＊Ｒ

○ラインの追加は次のようにおこないます。

（例） プログラム１を計算機に入れるとき、次のように　S＝π＊R＊R　をぬかした場合。

```
10  INPUT  R
                ←この間に　S＝π＊R＊R　を追加したい。
20  PRINT  S
30  END
```

10ラインと20ラインの間に追加する場合は、この間のラインナンバー（11～19のいずれか）を使ってプログラムを書きます。

たとえば　15　S＝π＊R＊R [ENTER] とします。そうすればプログラムは次のようになります。（[↓] キー及び [↑] キーで確認してください。）

```
10  INPUT  R
15  S＝π＊R＊R
20  PRINT  S
30  END
```

注）プログラムは若いラインナンバーから書く必要はなく、計算機が自動的に若いラインナンバーから順に並べかえてくれます。

注）ラインナンバーだけを変えたいときは [DEL ◄] キーでラインナンバーを点滅させて、そのナンバーを変更します。

（例） ３０　ＥＮＤ　のラインナンバーを４０に変えるとき

[◄] を操作　　３０　ＥＮＤ
　　点滅させます。

４ [ENTER]	４０：ＥＮＤ	40ラインが書き込まれます。
３０ [ENTER]	>	30ラインを削除します。

- この例のように、すでに書き込まれているラインのラインナンバーを変えて書き込んだ場合、変える前のラインはそのまま残っていますので、不要な場合は削除する必要があります。
- あるラインをすべて削除するときは ラインナンバー [ENTER] と押します。

○プログラムの内容の確認は次の方法でおこなってください。

① [↓] キー

このキーを１回押すごとに現在表示しているラインの次のラインを表示します。また、このキーを押し続けますと、順次、次のラインを表示します。

② [↑] キー

このキーを１回押すごとに現在表示しているラインの前のラインを表示します。また、このキーを押し続けますと、順次前のラインを表示します。

③ ＬＩＳＴ命令(リスト)

ＰＲＯモードで **ＬＩＳＴ** [ENTER] と押せば、もっとも若いラインを表示します。

ＬＩＳＴ ＿＿＿＿ [ENTER]
　　ラインナンバー

と押せば、指定されたラインを表示します。(もし、指定されたナンバーのラインがない場合は、それよりも大きく、かつ一番近いラインが表示されます。)

●ＬＩＳＴ命令はＰＲＯモードでのみ有効です。

注) 表示部には最大16桁表示することができますが、１ラインに長いプログラムを書きますと、その一部は表示できなくなります。このような隠れた部分を見るときは [►] キーを押し続けてください。そうすればカーソルが右端まで行き、そののち表示が左に送られて隠れている部分が出てきます。左に隠れた部分を見るときは [◄] キーを押します。

3-3）プログラムの実行（ＲＵＮ）

○プログラムの実行はＲＵＮモードでおこないますので、電源／モードスイッチを**ＲＵＮ**位置にしてください。

○続いて次の操作により実行を開始します。

ＲＵＮ [ENTER]　　(最も若いラインナンバーより実行)

ＲＵＮ ＿＿＿＿ [ENTER]　　(指定したラインナンバーより実行)
　　ラインナンバー

(例) プログラム１(23ページ)を実行しましょう。まずプログラムを計算機に入れてから次の計算をしてください。

①半径が５cmの場合

②半径が10cmの場合

	ＲＵＮモード	＞
①	ＲＵＮ [ENTER]	？
	５	５＿
	[ENTER]	７８．５３９８１６３４
	[ENTER]	＞

② RUN [ENTER]　　?
10　　10_
[ENTER]　　314.1592654
[ENTER]　　>
└プロンプト記号は、プログラムの実行が終ったことを示します。

○例のようにＩＮＰＵＴでは？を表示して実行をいったん停止しますので、データを入力して [ENTER] キーで実行を再開させます。
また、ＰＲＩＮＴ命令で、計算結果を表示しているときも、プログラムの実行をいったん停止しますので、[ENTER] キーで実行を再開させます。

＝エラー表示と処理方法＝

プログラムを実行しているときにエラーが発生しますと、

```
ERROR 3 IN 30
```

のように、エラー番号と、エラーが発生したラインナンバーを表示します。
このときは、次のようにしてエラー処理をおこなってください。

① [CL] キーでエラーを解除してください。
（エラーが表示されているとき、あるいは [CL] でエラーを解除したとき、[↑] キーを押せば、押している間だけエラーが発生したプログラムラインを表示します。）
② ＰＲＯモードにしてください。
③ [↓] キーあるいは [↑] キーを押してください。エラーが発生したプログラムラインが表示され、エラーが発生した位置がカーソルで示されています。
④ エラーが発生した原因を探して、26～27ページおよび134ページに記載の方法で訂正などをおこなってください。

- 電源／モードスイッチを切り替えれば、[CL] を押さなくてもエラーは解除されます。
- エラーの内容については133ページを参照してください。

4．プログラム（その2）

プログラム（その1）では、プログラムの構成や、編集などについて説明しました。
ここでは、基本的な入力命令、出力命令、およびジャンプ、判断、くり返しなど、プログラムにもっとも多く使われる命令について説明します。

4-1）入力命令（INPUT）

INPUT命令は、プログラム実行中にマニュアル操作により数値や文字の入力をおこなうための命令です。

(1) **INPUT変数，変数……**

```
INPUT  A              （変数Aに数値を入れなさい）
INPUT  C$             （変数C$に文字を入れなさい）
INPUT  D, E$, F       （複数の入力の指定）
```

(2) **INPUT"文字"，変数，"文字"，変数……**

```
INPUT"A=", A
INPUT"NAMAE=", B$, "TOSI=", C
```

INPUT時にメッセージを表示させる方法です。

（例）

```
10  INPUT  "A=", A
20  INPUT  "NAMAE=", B$,
       "TOSI=", C
    ⋮
```

○計算機に書き込んで実行してみてください。

```
RUNモード

RUN [ENTER]          A=
12                   12_
    [ENTER]          NAMAE=
TANAKA               TANAKA_
    [ENTER]          TOSI=
20                   20_
```

このように、この形の場合はデータを入力すればメッセージは消えます。

(3) INPUT"文字";変数,"文字";変数…

```
INPUT"A=";A
INPUT"NAMAE=";B$,"TOSI=";C
```

この形は(2)と同じようにINPUT時にメッセージを表示させる方法ですが、(2)とちがって、データを入力したとき、メッセージは消えず、メッセージの後に入力した内容が表示されます。

○ (2)で書き込んだプログラムを次のように変更して実行してみてください。

```
10 INPUT"A=";A
20 INPUT"NAMAE=";B$,
   "TOSI=";C
```

この部分を,(コンマ)から;(セミコロン)に変えています。

```
RUNモード

RUN [ENTER]          A=_
12                   A=12_
    [ENTER]          NAMAE=_
TANAKA               NAMAE=TANAKA_
    [ENTER]          TOSI=_
20                   TOSI=20_
```

上記(1)、(2)、(3)を混合して使うこともできます。

```
INPUT A,"B=",B,"C=";C
```

＝参　考＝

ＩＮＰＵＴ命令に続くステートメントのスキップ。

プログラムがＩＮＰＵＴ命令によるデータ入力待ちになっているとき、データなどの入力をおこなわずに ENTER キーを押しますと、いま実行しているところより後に書かれている命令などをスキップして（無視して）次のラインへ実行が移ります。
たとえば、次のような場合

```
      ⋮
20：INPUT　C：A＝A＋1：GOTO　10
30：
      ⋮
```

Ｃへの入力待ちになっているとき、データを入力せずに ENTER キーを押しますと、Ａ＝Ａ＋１：ＧＯＴＯ　10をスキップして、30ラインへ実行が移ります。

4-2）出力命令（ＰＲＩＮＴ）

ＰＲＩＮＴ命令は、計算の処理結果などを表示させるための命令です。

◉ **ＰＲＩＮＴは基本的に次の使いかたができます。**

(1) ＰＲＩＮＴ式

この形では式の値を表示部の右側にツメて表示します。

（例）　ＰＲＩＮＴ　Ｓ　　Ｓの内容を右ヅメで表示

注）式とは、数値や数値変数およびこれらを含んだ式をさします。

（例）　３，Ａ，３＊４，３＊Ａ＋５など

(2) ＰＲＩＮＴ文字変数

ＰＲＩＮＴ“文字”

この形では文字あるいは文字変数の内容を表示部の左端から表示します。

（例）　ＰＲＩＮＴ　Ａ＄　　Ａ＄の内容を左側より表示

ＰＲＩＮＴ　＂ＭＥＮＳＥＫＩ␣Ｓ＝＂　文字を左側より表示

←本書ではスペースを意味しています。計算機に入れるときは SPC キーを操作してください。

(3) ＰＲＩＮＴ { 式 / “文字” / 文字変数 } , { 式 / “文字” / 文字変数 }

16桁の表示部を左右８桁に分けて，,の左右の内容を表示させます。このときも、８桁の範囲内で数値は右ヅメ、文字は左ヅメにします。なお、表示内容が８桁を超える場合、文字は頭から８文字のみを表示し、数値は下位桁を切捨てて表示します。

ただし、数値が、-10^6以下または10^6以上のときは浮動小数点方式（指数方式）で表示され、絶対値で1より小さい数値は浮動小数点方式と固定小数点方式が自動的に選択され、表示されます。

（例）ＰＲＩＮＴ　Ａ，Ｓ　　Ａの値を左８桁内に、Ｓの値を右８桁内に右ヅメで表示。

ＰＲＩＮＴ〝Ｓ＝〟,Ｓ　　Ｓ＝を左８桁内に左ヅメで、Ｓの値を右８桁内に右ヅメで表示。

(4) ＰＲＩＮＴ { 式 / “文字” / 文字変数 } ; { 式 / “文字” / 文字変数 } ; … ; { 式 / “文字” / 文字変数 }

この形では文字や数値を表示部の左端よりツメて16桁の範囲内で表示します。

（例）ＰＲＩＮＴ　〝Ｓ＝〟；Ｓ　　左端にＳ＝を表示し、続いてＳの内容を表示します。

式とは

本書において式とは、数値や数値変数およびこれらを含んだ計算式をさします。

（例）　３，Ａ，３＊４，ＳＩＮ　Ｂ　など

4-3）代入文（ＬＥＴ）

代入文は、変数に数値や文字を代入するための命令で、LET（レット）に続いて代入式を書きます。

ＬＥＴ　Ｓ＝π＊Ｒ＊Ｒ

代入文は、通常ＬＥＴを省略して使用できます。ただし、後で説明しますＩＦ文に続いて書く場合はＬＥＴの省略はできません。

代入式は一般の数学とは違い、右辺の数値あるいは文字を左辺の変数に入れる（代入）することを意味します。したがって左辺は変数に限ります。

Ａ＝Ａ＋２　　（Ａの内容に２を加えてＡに入れなさい）

Ｂ$＝〝ＫＯＴＡＥ〝　　（Ｂ$にKOTAEという文字を入れなさい）

4-4）END文

BASICではプログラムの終りに“これでプログラムは終了です↙”という意味でEND文を書きます。

4-5）ジャンプ命令（GOTO）

ジャンプは“飛ぶ”ということで、プログラムの実行を指定したラインに飛ばすことです。通常プログラムは若いラインから順次実行されますが、GOTO命令（ゴートゥー）を実行しますと、GOTOに続いて書かれているラインへジャンプして、ジャンプしたラインから実行します。

```
GOTO  40      ラインナンバー40へジャンプ
GOTO  5*8     ラインナンバー40へジャンプ
GOTO  A       Aの内容のラインへジャンプ（A＝40ならば40ラインへジャンプ）
GOTO  "A"     "A"ラベルのついているラインへジャンプ
```

（例1）

```
10:INPUT "HANKEI=";R
20:S=π*R*R
30:PRINT "MENSEKI=";S
40:GOTO 10
```

40ラインから10ラインへジャンプします。

（例2）

```
10:"A": INPUT "HANKEI="
   ;R
20:S=π*R*R
30:PRINT "MENSEKI=";S
40:GOTO "A"
```

40ラインからラベル"A"へジャンプします。
このようなジャンプをラベルジャンプといいます。

ラベル：ジャンプ先につけた名前のことで、GOTO命令などでラベルジャンプが指定されていますと、実行の際、そのラベルを探してそのラインにジャンプします。ラベルジャンプの場合、プログラムの訂正などでラインナンバーをつけかえても、プログラムは変更する必要がないので便利です。なお、同じラベルが2個以上書かれているときは、ラインナンバーの小さいほうへジャンプします。

（注）（例1）、（例2）のプログラムは10ラインから40ラインまでをくり返し実行するようになっていますので、プログラムの実行が終了しません。このようなプログラムにおいて実行を終了させるときは ON BRK CL と押してください。

4-6）判　断（ＩＦ）

ＩＦ命令はいろいろな判断をおこなう命令です。たとえば

ＩＦ　Ａ>=10

の例では、Ａの値が10以上（ＩＦ文が成立）ならばＩＦ文の後に続く内容を実行し、10未満（ＩＦ文が不成立）ならば次のラインを実行します。

ＩＦ文に続く、Ａ>=10などの式を条件式といいます。　条件式は通常次のような形で書きます。

ＩＦ　○○=××　　等しいかどうか判断（○○は××に等しいか？）
ＩＦ　○○>××　　大きいかどうか判断（○○は××より大きいか？）
ＩＦ　○○>=××　以上かどうか判断（○○は××以上か？）
ＩＦ　○○<××　　小さいかどうか判断（○○は××より小さいか？）
ＩＦ　○○<=××　以下かどうか判断（○○は××以下か？）
ＩＦ　○○<>××　等しくないかどうか判断（○○と××は等しくないか？）
（注）　○○および××は式、変数を表わします。（５＊４、Ａ、8など）

ＩＦ文の後は通常ＴＨＥＮ60のように使い、指定したライン（60ライン）へのジャンプを指示します。（ＴＨＥＮ：ゼン）

```
ＩＦ　Ａ<１０　ＴＨＥＮ　６０
```

（Ａが10よりも小さいときは60ラインへジャンプ。Ａが10よりも大きいか等しいときは次のラインを実行。）

また、ＩＦ文の後にステートメントを続けることもできます。

```
IF B>=5 LET C=1200*B : GOTO 70
```

（Ｂが５よりも大きいか等しいとき、1200＊Ｂの結果をＣに代入して、70ラインへジャンプ。Ｂが５よりも小さいときは次のラインを実行。）

注）ＩＦ文に続いて代入文を書く場合は、必らずＬＥＴ命令が必要です。

条件式を通常の式にすることも可能です。たとえば

ＩＦ　Ａ>Ｂ+４　ＴＨＥＮ　６０

とした場合、Ａの値がＢ＋４の式の値よりも大きいならば60ラインを実行し、Ａの値がＢ＋４の式の値よりも大きくないならば、次のラインを実行します。

また、

ＩＦ　Ａ　ＴＨＥＮ　６０

のようにすることも可能で、この場合はＡの値が正（ＩＦ文が成立）ならばＩＦ文の後に続く内容（60ラインへのジャンプ）を実行し、Ａの値が０および負（ＩＦ文が不成立）ならば次のラインを実行します。

＝ご注意＝

本機は仮数部が最大10桁までの計算をおこなうことができますが、内部では計算の精度を上げるために仮数部12桁まで計算をおこない。11桁目を四捨五入処理したうえで計算結果しとて表示します。例えば5／9および5／9＊9の計算は内部で次のように計算されます。

5／9　→　5.55555555555E -01
　　└11桁目を四捨五入
表示　→　5.555555556E -01
5／9＊9→　4.99999999999E 00
　　└11桁目を四捨五入
表示　→　5.

このように内部では、あくまで仮数部12桁の数値を用いて計算がおこなわれています。
したがって、計算を連続しておこなった場合と、区切って計算した場合とでは、その結果に差が生じることがあります。

（例）　$3^2-9=$

〈連続した計算〉

3 [SHIFT] [^] 2 [−] 9 [ENTER] → −9. E −11

〈区切った計算〉

3 [SHIFT] [^] 2 [ENTER] → 9.
[−] 9 [ENTER] → 0.

また、ＩＦ文においても、条件式が連続的な計算では、この差のためにプログラムが目的通りに働らかない場合も生じますのでご注意ください。

（例2）
```
10  INPUT  A
20  IF  A^2 >=9 THEN  50
  :
```
Aが3のとき、3^2が計算機内部では8.99999999991E 00 となるためＩＦ文が成立しません。

このようなときは、次のように変数を用いて〈区切った計算〉になるようにプログラムしてください。

```
10  INPUT  A
15  B=A^2                    } A^2の結果を変数に代入し、その変数を
20  IF  B>=9  THEN  50       } 用いて条件式を構成します。
  :
```

（べき乗計算は、$\log x$と10^x計算を用いて計算されるため、計算機内との差が出やすくなります。
A^B → $10^{B \log A}$ により計算）

4-7）くり返し計算（FOR…TO…STEP, NEXT）

同じ計算を数値だけ変えて何回もくり返しておこなうような場合は、ＦＯＲ－ＮＥＸＴ文が便利です。

ＦＯＲ　ＮＥＸＴ文（フォー、ネクスト）の一般形は次のようになります。

ＦＯＲ　数値変数＝初期値　ＴＯ最終値　ＳＴＥＰきざみ
〜
ＮＥＸＴ　数値変数

数値変数が初期値から始まって、指定されたきざみずつ増加していき、最終値になるまでＦＯＲとＮＥＸＴの間の計算をくり返し実行します。（このくり返し部分をＦＯＲ－ＮＥＸＴループと呼びます。）なお、（最終値－初期値）÷きざみが割り切れない場合は、最終値を越える最小の値（きざみが負の場合は逆）になるまでＦＯＲ－ＮＥＸＴループの実行をくり返します。

○ ＦＯＲとＮＥＸＴは必ず対にして使い、ＦＯＲの後の数値変数とＮＥＸＴの後の数値変数は同一の変数でなければなりません。

○ きざみが１のときはＳＴＥＰ１は省略できます。

○ 初期値、最終値、きざみを負の値にすることもできます。

（例）　１０　ＦＯＲ　Ａ＝０　ＴＯ　－９０　ＳＴＥＰ　－５

◉ **ＦＯＲ－ＮＥＸＴループの注意事項**

ＦＯＲ－ＮＥＸＴループを使う場合に注意すべきことがいくつかありますので次に説明します。

① 最終値およびきざみ（ステップ数）はそれぞれ次の範囲内でなければなりません。

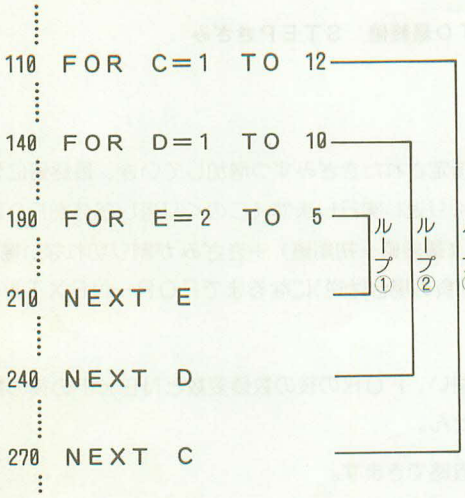
最終値：－32768〜32767
最終値に小数部が含まれる場合、小数部は無視されます。

きざみ：－32768〜32767　（ただし、きざみ≠０）
きざみの値に小数部が含まれる場合、小数部は無視されます。

（例）……ＳＴＥＰ　2.5　←　２がきざみ値になります。

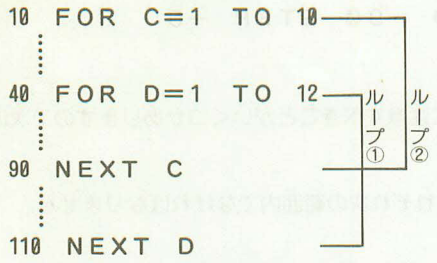
② FOR−NEXTループの中に、別のFOR−NEXTループを入れることができます。ただし、中に入るFOR−NEXTループは外のFOR−NEXTループ内に完全に入っていなければなりません。 この条件でループを5段まで重ねて使う（深みをもたせる）ことができます。

（例1）



それぞれ、中のループが外のループの中に完全に入っていますので、この使い方は正しい。
このような形でループ⑤まで作ることができます。

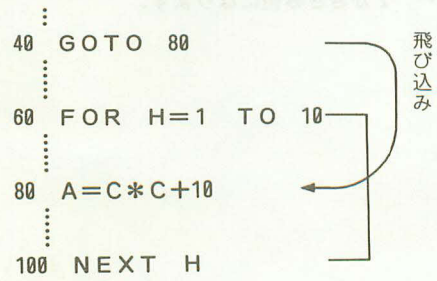
（例2）



ループ①とループ②が交差していますのでこの使い方は誤り（ERROR 5）

③ 外からFOR−NEXTループ内に飛び込むことはできません。

（例）



ループの外からループ内に飛び込むことはできません。（ERROR 5）

注）ＦＯＲ－ＮＥＸＴループから外に抜け出した場合は、ＦＯＲ－ＮＥＸＴループが終了したことにならず、ＦＯＲ文が何回か実行されるうちにエラー（ＥＲＲＯＲ 5）になります。

したがって、ＦＯＲ－ＮＥＸＴループを途中で抜ける場合は、カウント用の数値変数に最終値を代入したうえで、ＮＥＸＴ文の書かれているラインを実行させることにより、ＦＯＲ－ＮＥＸＴループを終了させてください。

（例）

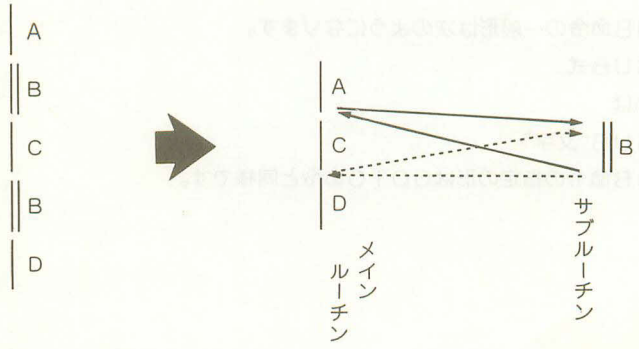
```
      ⋮
100   FOR  A=1  TO  30
      ⋮
150   IF  B=12  LET  A=30：GOTO  200
      ⋮
200   NEXT  A
      ⋮
```

4-8）サブルーチン（GOSUB、RETURN）

普通サブルーチンは、通常のプログラム（メインルーチン）から、必要に応じて呼び出され、実行されます。

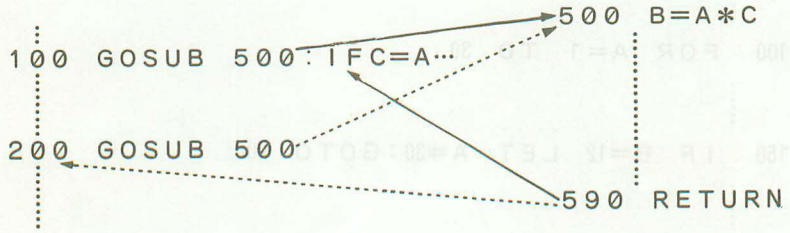
ある一連のプログラムで、何回も同じ計算や処理が出てくるとき、その部分をサブルーチンにすることにより、プログラムを短かく、また簡略化することができます。

たとえば、次の左図のように共通部（Ｂ）があるとき、その部分をサブルーチンにすると、右図のようになります。



サブルーチンへのジャンプは、GOSUB命令（ゴーサブ）によりおこないます。
サブルーチンのおかれているラインナンバー（あるいはラベル）をGOSUBの後に続いて書いて、サブルーチンジャンプを指示します。
また、サブルーチンの最後にはRETURN命令（リターン）を書いて、メインルーチンへの復帰を指示します。
メインルーチンへ復帰したときは、GOSUBのあった次の命令を引続き実行します。

〈例〉



●サブルーチンの中にサブルーチンを入れることもできます。この場合、10段まで深みをもたせることができます。

●GOSUB命令の一般形は次のようになります。

GOSUB式
あるいは
GOSUB"文字"

GOSUB命令の指定の形はGOTO命令と同様です。

5．変数

変数については2－5）でふれましたが、ここでは変数の構成などについて、さらにくわしく説明します。

5－1）変数の構成

- 変数
  - 数値変数
    - 数値固定変数（A～Z）
    - 数値配列変数
  - 文字変数
    - 文字固定変数（A$～Z$）
    - 文字配列変数

5－2）固定変数

これまでの説明やプログラム例で、変数をA、B、C…あるいはA$、B$、C$…と指定して使用しましたが、これらの変数を固定変数と呼びます。

固定変数は、変数として使用されるエリア（データ専用エリア）が独立して確保されており、このエリアにはプログラムなどが書き込まれることはありません。また、変数Aあるいは変数B…として使用される場所は次の図のようにすべて決っています。

〈データ専用エリア〉

26	ZあるいはZ$	
3	CあるいはC$	
2	BあるいはB$	
1	AあるいはA$	

変数名　　データの記憶場所

◉ **数値固定変数：A～Zの変数**

仮数部10桁、指数部－99～99で表わすことのできる数値を扱います。

◉ **文字固定変数：A$～Z$の変数**

文字固定変数は最大7字までの文字・記号を扱います。

5-3）配列変数（DIM）

配列変数は固定変数と違い、変数として使用される場所は決っていません。したがって、配列変数を使うときは、事前にその変数として使う場所をメモリー（プログラム・データエリア）の中に確保する必要があります。配列変数の場所を確保するためにはDIM（ディメンジョン）命令を使って、その変数名と大きさを定義します。

たとえばDIM　Z(9)は配列変数Z(　)のためにZ(0)からZ(9)までの10個の変数がメモリー内に確保されることを意味します。（下図参照）

Z(0)	Z(1)	Z(2)	Z(3)	Z(4)
Z(5)	Z(6)	Z(7)	Z(8)	Z(9)

◉ Z(0)、Z(1)…のように書かれている変数の、カッコの中の数字を添字といいます。本機では配列変数に添字が1個のものと2個のものを使用することができ、それぞれを一次元配列、二次元配列といいます。

一次元配列

DIM　Z(5)　　Z(0)、Z(1)……Z(5)の6個の変数を確保

二次元配列

DIM　Z(3、4)　Z(0，0)、Z(0，1)、Z(0，2)……Z(3，4)の20個の変数を確保（下図参照）

Z(0,0)	Z(0,1)	Z(0,2)	Z(0,3)	Z(0,4)
Z(1,0)	Z(1,1)	Z(1,2)	Z(1,3)	Z(1,4)
Z(2,0)	Z(2,1)	Z(2,2)	Z(2,3)	Z(2,4)
Z(3,0)	Z(3,1)	Z(3,2)	Z(3,3)	Z(3,4)

◉ **文字配列変数**

配列変数にも文字変数があり、固定変数と同様に$記号をつけて指定します。

（例）DIM　Z$(9)　　Z$(0)、Z$(1)……Z$(9)の10個の変数を確保
DIM　Y$(5,4)　Y$(0，0)、Y$(0，1)……Y$(5，4)の30個の変数を確保。

◉ **文字配列変数の拡張**

文字固定変数は、1個の変数に最大7文字まで記憶できる固定長の変数でしたが、配列変数では最大80文字以内で、任意に変数の長さを指定することができます。

（例） ＤＩＭ　Ｖ$（９）＊３０　**Ｖ$（０）、Ｖ$（１）、…Ｖ$（９）の変数にはそれぞれ最大30文字まで格納することができます。**

ＤＩＭ　Ｎ$（５，４）＊６　**Ｎ$（０，０）、Ｎ$（０，１）…Ｎ$（５，４）の変数にはそれぞれ最大６文字まで格納することができます。**

上例のように、ＤＩＭ文に ＊式 をつけて変数の長さを指定します。指定できる文字数は1～80文字で、指定しない（＊式がない）ときは自動的に16文字に設定されます。

◉ 配列変数はＸ（Ｂ）、Ｙ$（Ｃ）、Ｚ（Ｃ，Ｄ）のように、添字を変数や式で指定することもできます。

◉ 添字は理論的には０～255までの数値を用いることができます。
ただし、計算機のメモリー容量の大きさ、変数の長さ、またプログラムの大きさなどにより、添字で指定した数だけ、変数が確保できない場合があります。

◉ 配列変数をたくさん定義する場合は次のような形で一度に定義できます。

ＤＩＭ　Ｘ（２０），Ｙ（５，７），Ｚ$（１０，３）＊６

注1.同一プログラム内で、同じ変数名の配列を再定義することはできません。
ＤＩＭ　Ｘ（２０）とＤＩＭ　Ｘ（３，４）は同じＸ（　）という変数名になります。
もし、定義し直したい場合はＣＬＥＡＲ命令（46ページ参照）により、変数を消去してから、定義し直してください。
なお、数値配列変数と文字配列変数は別べつに確保されますので、たとえばＺ（　）とＺ$（　）を同時に使用することができます。

注2.配列変数の変数名はＢ（　）～Ｚ（　）およびＢ$（　）～Ｚ$（　）を使用してください。
なおＡ（　）およびＡ$（　）を使用する場合は特別な注意が必要ですので46ページをご覧ください。

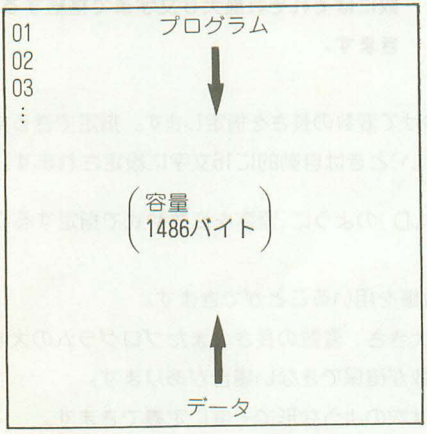
注3.ＲＵＮ命令によりプログラムを実行したとき、それまでにメモリー上に確保されていた配列変数は消去されます。（52ページを参照）
ＧＯＴＯ命令や定義付けキーによるプログラムの実行では、変数は消去されません。
したがって一度実行したプログラムを再実行するとき、ＤＩＭ命令の書かれているラインを実行させると、同じ変数名を再定義することになり、エラー（ＥＲＲＯＲ　３）になります。

5-4）メモリーの構成と変数

変数については2－5）で少しふれましたが、ここでは本機の持つメモリーの構成と、変数についてさらにくわしく説明します。

◉ **メモリーの構成**

〈プログラム・データエリア〉



プログラムがこちら側から書き込まれていきます。

C(1)やB(5，8)のような配列変数はこちら側から確保されます。

〈データ専用エリア〉(容量208バイト)

26	ZあるいはZ$	
25	YあるいはY$	
⋮	⋮	
2	BあるいはB$	
1	AあるいはA$	

固定変数
(26メモリー)

メモリーは前記のような構成になっています。プログラム・データエリアはプログラムメモリー、データメモリー(配列変数)のいずれにも使用することのできるメモリーです。図で示すように、プログラムと変数はそれぞれ逆方向から書き込まれ、確保されていきます。これに対してデータ専用エリアは常にデータメモリー(変数)として確保されています。通常、データメモリー(変数)として使用できるのはこの2種類のメモリーエリアですが、プログラム・データエリアを使用するときは書き込まれているプログラムの長さによって変数として使える大きさが変ってきますので注意が必要です。

次に各変数を確保する場合に使用するバイト数およびプログラムの各命令などの占めるバイト数を記しておきますので参考にしてください。

変　　数	変数の名前	デ　ー　タ
数値配列変数	6 バ イ ト	8 バイト
文字配列変数	6 バ イ ト	指定されたバイト数※

※ たとえば　ＤＩＭ　Ｚ$（２，３）＊10 と指定した場合は、10文字を記憶できる変数を12個確保しますが、これに使用するバイト数は６バイト（変数名）＋10バイト（文字数）×12個＝126バイトになります。

構　成　要　素	ラインナンバー	命 令 文、関 数	その他
使用バイト数	2バイト	1バイト	1バイト

＝残りバイト数と使用できる変数の数＝

プログラム・データエリアの残りバイト数は

ＭＥＭ ENTER

の操作で確認することができます。

また、ＤＩＭ命令で確保できる数値変数の数は次の計算で得ることができます。

（ＭＥＭ－6）／8 ENTER

文字変数の数を計算する場合は、
この８を文字数（指定する文字数）
に変えてください。

●ＭＥＭ命令については60ページを参照ください。

5-5）変数をクリア(消去)するには(CLEAR)

◉ 個々の変数をクリア(消去)するには次のようにします。

(例) A＝0
B(1)＝0 } 数値変数は0を代入することによりクリアします。

A$＝" "
C$＝" " } 文字変数は" " を代入することによりクリアします。

(" " を代入するということは文字変数を文字がない状態にするということです。
この状態をＮＵＬ（ヌ ル）あるいはＮＵＬ状態と呼びます。(ＮＵＬ：ＮＵＬＬ))

◉ ＣＬＥＡＲ命令(クリア)

すべての変数を一度にクリアするには、ＣＬＥＡＲ命令を実行します。ＣＬＥＡＲ命令の一般形は次のようになります。

ＣＬＥＡＲ

この命令が実行されれば、ＤＩＭ命令により確保されていた配列変数は消去され、固定変数の内容もすべて消されます。

●この命令ではプログラムはクリアされず、保護されます。

5-6）変数A(　)について

固定変数はデータ専用エリアを使用していますが、このデータ専用エリアを配列変数と同様の添字つき変数として指定することができます。

固定変数はＡ～Ｚ(Ａ$～Ｚ$)の26本あります。これにそれぞれ１～26の番号を割り当て、A(1)～A(26)あるいはＡ$(1)～A$(26)とすることにより、添字つき変数として指定されます。すなわち、A(1)はＡと同じであり、A(2)はＢと同じ変数になります。

◉ 添字を27以上にした場合

A(　)の形で添字を27以上にした場合は、プログラム・データエリアにその変数が確保されます。

たとえば　A(35)＝5　を実行すれば、プログラム・データエリアにはA(27)～A(35)の変数が確保されます。

A(27) 以上の変数は配列変数として扱われますが、ほかの配列変数と違って次のような注意が必要です。

1. ＤＩＭ命令により事前に変数を確保しなくても使用できます。
ただし、次の例のように配列が分かれるような場合はエラーになります。

（例） ＤＩＭＢ（３） ENTER
Ａ（28）＝５ ENTER
ＤＩＭＣ（２） ENTER
Ａ（30）＝９ ENTER ←エラー

上の操作をおこなうと右の図のような形で配列変数が確保され、Ａ（　）の配列が２つに分かれるためエラー（ＥＲＲＯＲ ３）になります。

〈プログラム・データエリア〉

Ａ（30）
Ａ（29）
Ｃ（２）
Ｃ（１）
Ｃ（０）
Ａ（28）
Ａ（27）
Ｂ（３）
Ｂ（２）
Ｂ（１）
Ｂ（０）

2. ＤＩＭ命令により変数を確保することもできます。ただし、
- 二次元配列にしたり、格納文字数の指定はできません。
Ａ＄（　）は最大７文字まで記憶できる固定長の変数になります。
- ＤＩＭＡ（26）のように添字が26以下でのＤＩＭ指定は無視されます。
- 既にＡ（27）以上の変数が使用されているとき、Ａ（27）以上の変数をＤＩＭ指定しますとエラーになります。

3. 同じ添字の数値変数と文字変数は共存できません。
たとえば、Ａ（30）とＡ＄（30）は同じ場所を使用することになりますので、同時に使用することはできません。

6．定義付けプログラム

本機はプログラム・データエリアの容量内であれば、何本でもプログラムを書き込むことができます。
そして、プログラムが複数本書き込まれているときに、2番目以降のプログラムを実行する場合は、次の方法でスタートさせることができます。

① RUN　ラインナンバー [ENTER]
② GOTO　ラインナンバー [ENTER]

この方法により、指定したラインからプログラムの実行が開始されます。この方法では操作に手間がかかり、また、ラインナンバーを常に知っていなければならず、なかなかわずらわしいものです。このようなとき、プログラムラインの始めに"A"や"B"のようにラベルを書いておくことにより、[DEF] [A] や [DEF] [B] と押すだけでプログラムの実行を開始させることができます。

定義付けとは、プログラムを下に示します18のキー（定義付けキー）のどれかに割り当てることです。
これらのキーに対応する文字・記号をラベルとしてラインナンバーのすぐ後に書き込んでおけば、[DEF] [A] 、[DEF] [S] 、……と押したときに、そのキーに対応するラベルを探して、そのラベルの書かれているラインからプログラムが実行されます。

（例）

```
        ラベルA
 50 : "A" : INPUT A
        ラベルS
120 : "S" : N=0 : T=0
        ラベルD
210 : "D" : CLEAR
```

◉ **定義付けキー**

定義付けのできるキーは次の18キーです。

[A] 、[S] 、[D] 、[F] 、[G] 、[H] 、[J] 、[K] 、[L] 、[=]
[Z] 、[X] 、[C] 、[V] 、[B] 、[N] 、[M] 、[SPC]

注）定義付けキーによりプログラムの実行開始を指定したとき、そのラベルをつけた（定義付けされた）プログラムがない場合はエラーになります。このときは [CL] キーでエラーを解除してください。（[►] 、[◄] キーでも解除できます。）

7．表示に関するＢＡＳＩＣ

7-1）フォーマット指定(USING)

ＵＳＩＮＧ命令（ユージング）は数値などを表示するときに、その表示フォーマットを指定します。

○ USING命令の一般形

① ＵＳＩＮＧ　"＃＃＃"　　符号を含めて整数３桁を表示

② ＵＳＩＮＧ　"＃＃＃."　　符号を含めて整数３桁と小数点を表示

③ ＵＳＩＮＧ　"＃＃＃．＃＃"　符号を含めて整数３桁と小数点と小数点以下２桁表示

④ ＵＳＩＮＧ　"＃＃．＃＃^"　小数以下２桁までの指数方式での表示

（このとき、仮数部の整数は符号を含めて２桁、指数部は符号を含めて４桁、（Ｅ−ＯＯ）が自動的にとられます。）

⑤ ＵＳＩＮＧ　"＆＆＆＆＆＆"　文字を６桁表示

⑥ ＵＳＩＮＧ　"＃＃＃＆＆＆＆"　数値と文字を同時に指定

⑦ ＵＳＩＮＧ　　フォーマット指定を解除

○ＵＳＩＮＧ命令に続く＂　＂内の記号はそれぞれ次の指定をおこないます。

＃	数値の桁数を指定 整数部桁数：２〜11桁（符号を含む） 指定桁より数値のほうが少ないときはその分だけスペース表示になり、多いときはエラーになります。 小数部桁数：０〜12桁（浮動小数点表示のときは０〜９桁） 指定桁より数値のほうが少ないときはその分だけ０表示になり、多いときはその分だけ切捨てられます。
．	小数点の表示を指定（整数部と小数部の区切りを指定）
^	数値の浮動小数点表示（指数方式）を指定 整数部の指定桁にかかわりなく、仮数部の整数桁は２桁（符号を含む）になります。 小数部の桁数を９桁以上に指定した場合、仮数部の小数桁は９桁になります。
＆	文字列の桁数を指定 指定桁より文字数が少ない場合はその分スペース表示になり、文字数が多い場合は指定された桁分だけ表示します。

○USINGを単独に使います。

```
10:A=12.3,B=-34.5678
20:USING "####.###"
30:PRINT A;B
40:END
```

表示例

```
  12.300  -34.567
```

␣スペース ␣スペース ␣スペース

○ PRINT文の中にUSING文を含め、PRINT USING文として使います。

```
10:B=-10,C=10.7703
20:PRINT USING "&&&###"
   ;"B";B;" C="; USING
   "###.###";C
30:END
```

表示例

```
B  -10  C= 10.770
```

␣スペース ␣スペース ␣スペース ␣スペース

○USINGの解除

USING

USINGで表示フォーマットの指定をおこないますと、以降に実行されるPRINT命令、PAUSE命令には、すべてそのフォーマット指定が有効になりますので、不必要なときは本命令によってフォーマット指定を解除しておきます。

=注=

USING命令により、表示フォーマットが指定されている場合、PRINT命令の形によっては指定されたフォーマットで表示できなくなり、エラー（EORRR 7）になります。（100ページを参照ください。）

① 「PRINT 式」の場合は、USINGで指定された総桁数が16桁を越えているとエラーになります。
ただし、整数部の指定桁が11桁を越えているときは、越えた部分が無視されます。

16桁

```
10 USING "###############.####"
```

11桁

↑11桁を越えた部分は無視されます。

この指定の場合、総桁数は16桁とみなされるため、エラーになりません。

② 「ＰＲＩＮＴ　式，式」の場合は、ＵＳＩＮＧにより固定小数点方式が指定されているとき、整数部(符号、小数点を含む)の桁数が８桁を越えているとエラーになります。

③ 「ＰＲＩＮＴ　式；式」の場合は、エラーになりませんが、表示される桁は左から16桁目までです。

7-2）表示時間の指定（ＷＡＩＴ）

ＷＡＩＴ命令(ウェイト)は、ＰＲＩＮＴ命令によるプログラムの停止時間を指定する命令です。ＷＡＩＴ命令による指定はＰＲＩＮＴ命令のすべての形に対して有効で、プログラムは指定された時間停止し、その後、自動的に実行を再開します。

○ 停止時間は次のように指定します。

ＷＡＩＴ式

式の値により、決められている時間だけプログラム実行を停止して内容を表示し、その後プログラムの実行を再開します。

式の値は０～65535まで指定できます。なお、式の値１は約1/64秒に相当します。

○ ＷＡＩＴ指定の解除

ＷＡＩＴ

ＷＡＩＴでの指定は、以降のＰＲＩＮＴ命令に対してすべて有効となるため、時間を無限にしたいときなどは、本命令によって時間指定を解除してください。

7-3）ＰＡＵＳＥ命令

ＰＲＩＮＴ命令が表示をおこなってプログラムの実行を停止、あるいはＷＡＩＴ命令により停止時間が指定されるのに対して、このＰＡＵＳＥ命令（ポーズ）は停止時間が一定（約0.85秒）に決っており、その時間だけ表示をおこなって次のプログラムに実行が移るところが異っているだけで、そのほかはＰＲＩＮＴ命令と同じです。

=プログラムの実行開始方法とそのスタート状態の違いについて=

プログラムの実行開始方法には次の方法があります。

① RUN命令によるもの

RUN [ENTER]

RUN式 [ENTER]

RUN"ラベル" [ENTER]

② GOTO命令によるもの

GOTO式 [ENTER]

GOTO"ラベル" [ENTER]

●GOTO [ENTER] ではエラー(ERROR 1)になります。

③ 定義付けキーによるもの

[DEF] に続くA、S、D、F……

これらの開始方法により、変数のクリアや、状態の解除などに違いがありますので次に示します。

RUNによる実行	GOTOによる実行 定義付けキーによる実行
●表示フォーマット(USING)指定を解除	●表示フォーマット指定を保持
●ウェイト(WAIT)指定を解除	●ウェイト指定を保持
●配列変数をクリア※ (DIM指定を解除)	●配列変数を保持 (DIM指定を保持)
●FOR-NEXTスタック GOSUBスタックをクリア	●FOR-NEXTスタック GOSUBスタックをクリア
●READ文に対するDATAの初期化	●READ文に対するDATAの初期化はおこなわない
●PRINT=LPRINT指定を解除 (PRINT=PRINTを指定)	●PRINT=LPRINTあるいは PRINT=PRINTの指定を保持

※RUN命令によりプログラムを実行しますと、それまでメモリー(プログラム・データエリア)上に確保されていた配列変数(DIM指定)およびA(27)以上の変数は消去されます。

ただし、固定変数(A～Z、A$～Z$、A$(1)～A$(26))は保持されています。

8．そのほかのＢＡＳＩＣ

8-1）READとDATA、RESTORE（リードとデータ、リストア）

変数にデータを入れる(代入する)方法として、ＬＥＴやＩＮＰＵＴのほかにＲＥＡＤ命令とＤＡＴＡ命令の組み合わせがあります。
これはＲＥＡＤに続いて変数を書き、そのデータをＤＡＴＡに続いて書きます。この場合の一般形は次のようになります。

```
READ　変数，変数，……
  ⋮
DATA　式，式，……（DATA　文字列，文字列，……）
```

●ＤＡＴＡ文の中に、式と文字列を混ぜて書くこともできます。

（例）

```
  ⋮
 50：READ　A，B
  ⋮
110：DATA　3，4，5
120：DATA　60，"G=","H="
  ⋮
200：READ　C，D，E$，F$
  ⋮
```

Aには3、Bには4が代入されます。

Cには5、Dには60、Ｅ$には"G="、Ｆ$には"H="が代入されます。

○　上例のようにＲＥＡＤの後の変数と、ＤＡＴＡの後のデータは１対１で対応します。したがって変数の形とデータの形が一致していなければなりません。すなわち、数値変数に対応するデータは数値でなければなりませんし、文字変数に対応するデータは文字列でなければなりません。

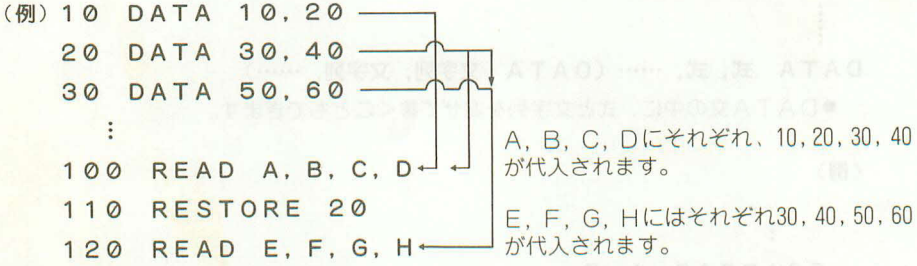
○　１つのプログラムの中に何回でもＲＥＡＤ文およびＤＡＴＡ文を書くことができますが、データは何ラインにわけて書いても一連のデータと見なされ、若いラインのデータから順番に変数に代入されます。
また、複数のプログラムが書き込まれ、それぞれのプログラムにＤＡＴＡ文がある場合でも、それは一連のデータと見なされます。
したがって、それぞれのプログラム内のデータを使用するときは、プログラムのスタートラインに、次で説明のＲＥＳＴＯＲＥ命令を書き込んでください。

◉ RESTORE命令（リストア）

READ命令の実行時、DATA命令で指定されているデータのどのデータを読み込むかは常に計算機に記憶されていますが、この読み込むデータの順番を強制的に変えるときに使用します。

○ RESTORE式（あるいはRESTORE “ラベル”）

この形の場合、この命令を実行した次のREAD命令の変数には、RESTORE命令の式の値（あるいはラベル）で指定されるラインに書かれているDATA命令の最初のデータから1対1に対応して代入されます。



○ RESTORE

この形の場合、この命令を実行した次のREAD命令の変数には、プログラムの最も若いラインに書かれているDATA命令のデータが1対1に対応して代入されます。

8-2）ON GOTO、ON GOSUB

◉ ON GOTO命令（オン・ゴートゥ）

GOTO命令についてはすでに述べましたが、ON GOTO命令はGOTO命令の機能を拡大したものです。 ON GOTO命令の一般形は次のようになります。

ON 式 GOTO式$_1$,式$_2$,式$_3$,……（ON 式 GOTO〝ラベル〟,〝ラベル〟,…）

この命令ではONの次の式の値が1の場合はGOTOに続いて書かれている式$_1$で示されるラインへジャンプし、ONの次の式の値が2の場合は式$_2$で示されるラインへとジャンプします。もし、式の値が、1より小さいときやGOTOに続いて書かれている式の数より大きい値のときは、ON GOTO命令の書かれている次のラインへ実行が移ります。

（例）

```
50：ON A GOTO 100, 110, 120
```

Aの値が1のときは100ラインへ、2のときは110ラインへ、3のときは120ラインへ、ジャンプします。
Aの値が、1、2、3以外のときは、次のラインへ実行が移ります。

注）GOTOに続く式には、二次元配列変数(B(1,2),B(5,3)など)を使用することはできません。

◉ **ON GOSUB命令(オン・ゴーサブ)**

ON GOTO命令と同じ形で指定すれば、ONに続く式の値により、GOSUBに続いて書かれた式あるいはラベルで示すラインへサブルーチンジャンプします。

8-3）REM(リマーク)

この命令は、プログラムの実行には関係なく、プログラムリストなどをわかりやすくする目的でプログラムの先頭や途中に注釈を入れておくものです。

（例）

```
10 REM KINRI_KEISAN
200 REM SUBROUTINE
```

8-4）STOP(ストップ)

この命令はプログラムの一時停止命令です。この命令を実行しますと「BREAK IN 200」のように、実行したラインナンバーを持ったブレークメッセージを表示して実行を停止します。

（例）

```
60：STOP
```

STOP命令により実行が一時停止している状態(ブレーク状態)のときはマニュアル操作が可能になるため、変数の状態を調べたり、変数に数値を代入したりする場合によくこの命令が使用されます。

○ STOP命令により停止した実行を再開させるときはCONT [ENTER] と操作してください。

8-5）AREAD（オートリード）

この命令は定義付キーにより、プログラムの実行を開始したとき、表示されていた内容を指定された変数に読み込む命令で、定義付プログラム(48ページ参照)で、ラベルに続いて書きます。

この命令の一般形は次のようになります。

AREAD　変数

（例）
```
10 "A":AREAD A
 ⋮
100 "B":AREAD A$
```

- この命令は、プログラム実行開始直後以外はスキップ(無視)されます。
- プログラムの実行を開始したとき、表示がプロンプト表示になっていたときは、指定されている変数はクリアされます。
- プログラムの実行を開始したとき、ＰＲＩＮＴ命令による表示がなされていたときは、次の内容が読み込まれます。

(たとえば次のようなプログラムを実行した場合)
```
10 "A":PRINT "ABC","DEFG"
20 "S":AREAD A$:PRINT A$
RUNモード
[DEF][A] → ABC      DEFG
[DEF][S] → DEFG
```

- ＰＲＩＮＴ式，式　あるいは　ＰＲＩＮＴ"文字"，"文字"により表示がなされていた場合は、右側に表示されていた内容が読み込まれます。
- ＰＲＩＮＴ式；式；式…により表示がなされていた場合は最初(いちばん左)に表示されていた内容が読み込まれます。
- ＰＲＩＮＴ"文字"；"文字"；"文字"…により表示がなされていた場合は最後に指定されている文字が読み込まれます。

9．特別な機能

9-1）乱数関数（RND，RANDOM）

◉ **RND命令（ランダム）**

RND命令は乱数を発生させる機能を持つ関数です。

RND　X　においてXの値により次のような乱数を得ることができます。

○Xが負数の場合：　直前に発生した乱数（あるいは乱数列）と同じ乱数を発生させるために、初期値を一定にします。

○Xが0から1未満の場合：0から1未満の乱数を発生します。

○Xが1以上の場合

Xが整数のとき：　1からXの値以下の乱数を発生します。
（1≦RNDX≦X）

Xが小数部を含むとき：　1からXの整数部に1を加えた値以下の乱数を発生します。（1≦RNDX≦（INT X）＋1
ただし、この場合は小数部の値によって乱数の発生に差が出ます。

（例）

```
10:DIM B(7)
20:FOR A=0 TO 1
30:D= RND -1
40:FOR B=0 TO 3
50:B(B+A*4)= RND 9
60:NEXT B
70:NEXT A
80:USING "##"
90:PRINT B(0);B(1);B(2)
   ;B(3);B(4);B(5);B(6)
   ;B(7)
100:END
```

←この代入文は同じパターンの乱数を発生させるために入れています。
このラインを削除すれば同じパターンの乱数は発生されなくなります。
（変数Dは、Dにかぎらず、プログラム内で他に使用していない変数であればよい。）

表示例

```
 9 7 1 3 9 7 1 3
```

4個の乱数を2回表示させています。

○乱数の有効桁数は10桁です。

◉ RANDOM命令（ランダマイズ）

RND命令の使用に先立って乱数のタネを植えつけるものです。

RNDは電源オンからやり直すと、常に同じ乱数を発生します。しかし、電源オン後にRANDOM命令を実行すれば同じ乱数を発生することは、まず、なくなります。

（例）

```
10：RANDOM
  ⋮
```

9-2）INKEY$（インキードル）

プログラムの中で、この命令が実行されたとき、いずれかのキーが押されていれば、その内容を読み込んで指定された変数に代入します。なお、このときのキー操作に対しての表示はおこなわれません。

INKEY$の一般形は次のようになります。

文字変数＝INKEY$

（例）

```
10:Z$= INKEY$
20:IF Z$="" THEN 10
30:PRINT "---";Z$;"---"
40:GOTO 10
```

20行：このラインをくり返し実行し、キーが押されるのを待ちます。

〔プログラムの実行を終了するときは ON BRK CA CL と押してください。〕

注1）プログラムのはじめにINKEY$がありますと、RUN ENTER などでプログラムをスタートさせたとき、ENTER キーを読み取ってしまうことがありますので注意が必要です。

注2）ENTER 、CL 、DEF 、SHIFT 、↓ 、↑ 、← 、→ キーがINKEY$に読み取られた場合、その内容はNUL（空白）になります。
なお、ON BRK キーはプログラムの一時停止キー（ブレークキー）として働きます。

注3）INKEY$命令は SHIFT キーや DEF キーが押されていれば、そのキーを読み込みます。したがって、SHIFT キーに続いて押したときに働く機能や入力される記号、DEF キーに続いて押したときに働く機能などを読み込むことはできません。

9-3）音発生機能（ＢＥＥＰ）

本機には音発生機能がついていますので、プログラムの途中で音を出し、計算機の操作をより効果的に、また楽しくすることができます。

◉ **ＢＥＥＰ命令（ビープ）**

ＢＥＥＰ命令は音の発生(発生回数)を指定する命令です。ＢＥＥＰ命令の一般形は次のようになります。

ＢＥＥＰ　式

式の値で示す数だけピ音を発生します。

式の値は整数部のみ有効で、正の数値でなくてはなりません。

（例）

```
      ⋮
50：BEEP　5　　←ピ音を5回発生
      ⋮
```

9-4）ＣＯＮＴ(コンティニュー)

ＣＯＮＴ命令は、プログラムの実行が一時停止しているときに、プログラムの実行を再開させるための命令です。したがって、この命令はプログラムに書き込むことはできず、ＲＵＮモードでのマニュアル操作でのみ有効となります。この命令の一般形は次のようになります。

ＣＯＮＴ　[ENTER]

○ プログラムの一時停止状態(ブレーク状態)とは次の状態です。

①プログラム実行中にＳＴＯＰ命令で一時停止した状態

②プログラム実行中に [ON BRK] キーを押して一時停止した状態

③そのほか、ＰＲＩＮＴ命令の実行などにより実行が停止している状態

9-5）MEM（メム）

プログラム・データエリアの中で空いている部分（プログラム、配列変数で使用されている部分以外）のバイト数を確認するための命令です。この命令の一般形は次のようになります。

MEM

（MEM [ENTER]
数値変数＝MEM）

9-6）論理積、論理和、否定（AND, OR, NOT）

◉ **論理積（ＡＮＤ：アンド）**

論理積は次のような値を取る関数です。

1 AND 1 ＝1
1 AND 0 ＝0
0 AND 1 ＝0
0 AND 0 ＝0

たとえば41と27の論理積（ＡＮＤ）を計算する場合は、これらの数値を２進数に変換して、それぞれの桁のＡＮＤを取り、その結果を10進数に変換します。

41 AND 27 ＝9

AND ＜ 101001——41
　　 011011——27
　　 001001—— 9

◉ 論理和（ＯＲ：オア）

論理和は次のような値を取る関数です。

1　OR　1　＝1
1　OR　0　＝1
0　OR　1　＝1
0　OR　0　＝0

たとえば41と27の論理和（ＯＲ）はＡＮＤ命令と同様、２進数に変換して、それぞれの桁のＯＲを取りその結果を10進数に変換します。

41ＯＲ27　＝59

OR＜ 101001――41
　　 011011――27
　　 111011――59

◉ 否定（ＮＯＴ：ノット）

10進数を16桁の２進で表わすと次のようになります。

10進	16桁の２進
32767	0111111111111111
⋮	
2	0000000000000010
1	0000000000000001
0	0000000000000000
−1	1111111111111111
−2	1111111111111110
⋮	⋮
−32768	1000000000000000

ここで２進数0000000000000001の否定（ＮＯＴ）を取りますと次のようになります。

ＮＯＴ（否定）　0000000000000001 → 1111111111111110

このように、各桁の１を０に、０を１に反転することを“否定（ＮＯＴ）を取る”といいます。さて、いま１とＮＯＴ１を加え合わせると、どうなるでしょうか。

　　0000000000000001　（１）
＋）1111111111111110　（ＮＯＴ１）
　　1111111111111111　（−１）

このようにすべての桁が１になります。これは上記の数値表で見ると、10進数では−１になります。

すなわち、１＋ＮＯＴ １＝－１となります。
このように数値Ｘとその否定（ＮＯＴ Ｘ）の間には

　　Ｘ＋ＮＯＴ Ｘ＝－１

の関係があります。
このことからＮＯＴ Ｘ＝－Ｘ－１
すなわち　ＮＯＴ Ｘ＝－（Ｘ＋１）
の関係式が導き出されます。
この関係式から

　ＮＯＴ ０＝－１
　ＮＯＴ－１＝０
　ＮＯＴ－２＝１

となることがわかります。

９-７）条件式の結合（ＡＮＤ，ＯＲ）

これまで、ＩＦ文の中で使用しました条件式は

　Ａ＞５
　Ａ$＝"Ａ"
　Ａ＊Ｂ＞＝Ｃ＋20

などのような、２つの値の大小比較の式が１つのみでしたが、これらの大小比較の式を２つ以上結合して、１つの条件式をつくることができます。たとえば“Ａは５よりも大きくかつ10よりも小さい”という条件を式で表わす場合、これまではＩＦ文を２つ使って

```
10　ＩＦ　Ａ＜＝５　ＴＨＥＮ　30
20　ＩＦ　Ａ＜10　ＴＨＥＮ　60
30
：
```

のような形で表わす必要がありましたが、ＡＮＤ命令を使えば

```
10　ＩＦ　Ａ＞５　ＡＮＤ　Ａ＜10　ＴＨＥＮ　60
```

のように１つの条件式にすることができます。

◉ **ＡＮＤ命令（アンド）**
２つ以上の条件のすべてを満足するような条件を１つの式で表わすときに用います。

（例）Ａは０よりも大きくかつ６より小さい場合の条件式

```
Ａ＞０　ＡＮＤ　Ａ＜６
```

◉ ＯＲ命令(オア)

２つ以上の条件のいずれかを満足するような条件を１つの式で表わすときに用います。

（例）Ａは０よりも小さいか、あるいは12より大きい場合の条件式

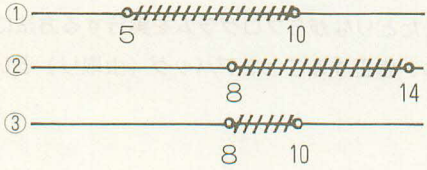
Ａ＜０　ＯＲ　Ａ＞12

○ＡＮＤ、ＯＲ、を使った条件式の例

① Ａ＞５　かつ　Ａ＜10

② Ａ＞８　かつ　Ａ＜14

③ ①と②のＡＮＤ



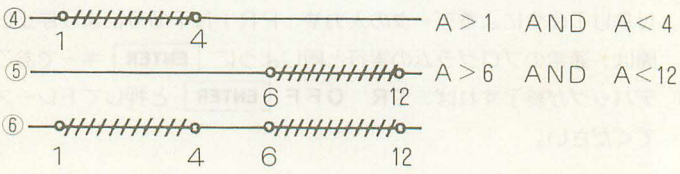
（Ａ＞５　ＡＮＤ　Ａ＜10）ＡＮＤ（Ａ＞８　ＡＮＤ　Ａ＜14）

この場合カッコはなくても同じです。

④ Ａ＞１　かつ　Ａ＜４

⑤ Ａ＞６　かつ　Ａ＜12

⑥ ④と⑤のＯＲ



（Ａ＞１　ＡＮＤ　Ａ＜４）ＯＲ（Ａ＞６　ＡＮＤ　Ａ＜12）

＝参　考＝

論理計算は次のように構成することにより、論理和（ＯＲ）、論理積（ＡＮＤ）を形づくることができます。

① 論理和　（論理計算）＋（論理計算）

例　（Ａ＜０）＋（Ａ＞８）　Ａが０より小さいか、あるいは８より大きいときに１の値を取る

（Ｂ＞０）＋（Ｃ＞０）　ＢあるいはＣが０より大きいとき、およびＢ、Ｃともに０より大きいとき１（あるいは２）の値を取る

② 論理積　（論理計算）＊（論理計算）

例　（Ｂ＞１）＊（Ｂ＜６）　Ｂが１より大きく、かつ６より小さいとき１の値を取る

③ 特殊な使い方の例

例　Ｂ＝（Ａ＝１）＊３＋（Ａ＝２）＊７　Ａが１のときＢは３になりＡが２のときＢは７になる。その他のときはＢは０になる。

10. デバッグ(TR ON、TR OFF)

プログラムを実行した場合に、何かの誤りで迷走したり、思わぬ結果が出てきたりすることがあります。
このような場合、プログラムのリストを調べて、誤りを探しますが、それだけではなかなか見つけにくい場合があります。
このようなとき、プログラムを１ラインずつ実行させ、その経過をたどりながらプログラムの誤りを探していけば、見つけやすくなります。
ここでは、１ラインずつ経過をたどりながらプログラムを実行する方法について説明します。（プログラムの誤りを探し、修正することをデバッグ（虫取り）といいます。）

◉ **デバッグのしかた**

(1) デバッグはＲＵＮモードでおこないますので、ＲＵＮモードにしてください。
(2) ＴＲ　ＯＮ [ENTER] と押し、トレースモードを指定します。
(3) ＲＵＮ [ENTER] と押してプログラムの実行を開始します。(最初のラインの実行が終われば実行は停止します。)
(4) その後は [↓] キーを押します。(１ラインだけ実行して停止します。) ただし、ＩＮＰＵＴ命令によるデータの入力や、ＰＲＩＮＴ命令により停止しているときの再開は、通常のプログラムの実行と同じように [ENTER] キーでおこないます。
(5) デバッグが終了すれば、ＴＲ　ＯＦＦ [ENTER] と押してトレースモードを解除してください。

次にごく簡単な例を示します。

〈プログラム例〉

```
10  INPUT"A=";A,"B=";B
20  C=A*2
30  D=B*3
40  PRINT  "C=";C;"␣D=";D
50  END
```

〈実行〉

RUNモード

TR ON [ENTER]	>	
RUN [ENTER]	A=_	} INPUT命令実行
8 [ENTER] （データ入力）	B=_	
9 [ENTER] （データ入力）	10：	←10ライン終了
[↓]	20：	←20ライン終了
[↓]	30：	←30ライン終了
[↓]	C=16. D=27.	←PRINT命令実行
[ENTER]	40：	←40ライン終了
[↓]	>	←実行終了
		（プロンプト表示）

○デバッグ動作でラインナンバーを表示しているときは、変数の内容をマニュアル操作によって呼び出し、所望の値になっているかどうかチェックできます。また、このとき [↑] を押せば、押している間止まっているラインの内容を表示します。（なお、[↑] キーを押して離したときは、プロンプト記号(＞)が表示されます。）

注）トレースモードは、TR OFF [ENTER] と押すか、[SHIFT] [CA CL] と押す、あるいはスイッチをOFF側にして電源を切るまで指定された状態が保たれます。

◉ **プログラムの途中で実行を停止させてチェックする場合**

○プログラムの途中にSTOP命令を書いておけば、プログラム実行途中でSTOP命令を実行し、ブレークメッセージを表示して実行が停止します。
このとき

①マニュアル操作で変数の内容をチェックする。
②トレースモードを指定して [↓] キーの操作にて以降のラインを1ラインずつデバッグ操作する。
③トレースモードを解除して CONT [ENTER] と操作し、通常の実行状態にて続行する。

○通常のプログラム実行中に [ON BRK] キーを押しますと、現在実行しているラインの終わりで実行を停止し、ブレークメッセージを表示します。このときも前記同様3つの操作をおこなうのが普通です。なお、ブレーク状態(停止状態)のときに [↑] を押せば、押している間停止しているプログラムラインが表示されます。

○トレースモードを指定しない場合でも、[ON BRK] キーやＳＴＯＰ命令などにより一時停止しているプログラムを [↓] キーにより１ラインずつ実行させていくことができます。この場合も表示はトレースモードが指定されている場合と同じようになります。

なお、[↓] キーを押したままにしますと、普通のプログラム実行時と同じように、プログラムが実行されます。

11. 文字の編集

この項では、文字を分断したり、結合したり、また、文字列の大小比較など、文字と文字の扱いについて説明します。

11-1）文字列の分断（ＬＥＦＴ$，RIGHＴ$，MID$）

◉ **ＬＥＦＴ$関数（レフトドル）**

この関数は、ある文字列の左側から何文字かを取り出す関数です。この関数の一般形は次のようになります。

ＬＥＦＴ$(文字変数,式)
あるいは
ＬＥＦＴ$("文字",式)

（例）

```
10:A$="ABCDE"
20:B$= LEFT$ (A$,3)
30:PRINT B$
```

Ａ$の文字列の左側３文字、←すなわちＡＢＣを取り出し、Ｂ$に代入します。

◉ **RIGHＴ$（ライトドル）**

この関数はＬＥＦＴ$関数とは逆に、右側から何文字かを取り出す関数です。一般形は次のようになります。

RIGHＴ$(文字変数,式)
あるいは
RIGHＴ$("文字",式)

（例）

```
10:A$="ABCDE"
20:B$= RIGHT$ (A$,3)
30:PRINT B$
```

Ａ$の文字列の右側３文字、←すなわちＣＤＥを取り出し、Ｂ$に代入します。

◉ MID$(ミッドドル)

この関数は文字列の中間の文字だけを取り出す関数です。この関数の一般形は次のようになります。

左側何文字目から取り出すか指定

MID$(文字変数,式$_1$,式$_2$)

何文字を取り出すか指定

MID$("文字", 式$_1$, 式$_2$)

(例)

```
10:A$="ABCDE"
20:B$= MID$ (A$,2,3)
30:PRINT B$
```

←A$の文字列の左側2文字目から3文字、すなわちBCDを取り出し、B$に代入します。

11-2) 文字列の結合

いくつかの文字列を結合させて、新しい文字列を作る場合は、"文字列の加算"をおこないます。たとえば、A$="ABC"、B$="DEF"、"XYZ"の3つの文字列を結合させる場合は

A$+B$+"XYZ"

のように加算の形で指定することができます。

注) 通常、文字固定変数およびA$( )の形の文字変数は7文字まで記憶することができ、それ以外の文字配列変数は特に指定していない場合は16文字まで記憶することができます。
文字列を変数に代入する場合はこれらの記憶可能文字数を越えないようにしてください。もし、越える可能性がある場合はDIM指定により文字配列変数の大きさを広げてお使いください。

11-3) 文字数を数える方法(LEN)

1つの文字列の中に含まれる文字の数(記号、スペース、数字も含みます)は計算機により、たちどころに求めることができます。

◉ **ＬＥＮ関数(レングス)**

たとえば

ＡＬＥＮ "ＡＢＣ１２５６"

のように指定して実行すれば "ＡＢＣ１２５６" の文字数7が変数Ａに代入されます。

この関数の一般形は次のようになります。

ＬＥＮ { “文字” / 文字変数 }

11-4）数値と文字の変換(ＶＡＬ，ＳＴＲ$)

数値を文字列として扱いたい場合や、逆に文字列として指定されている数字を数値として扱いたい場合がありますが、このようなとき、ＳＴＲ$あるいはＶＡＬ関数を使います。

◉ **ＳＴＲ$関数(ストリングドル)**

この関数は数値を文字列の形に変換する関数です。たとえば、Ｂ＝1234のとき

Ａ$＝ＳＴＲ$　Ｂ

として実行すれば、Ａ$には"1234"という文字列が代入されます。この関数の一般形は次のようになります。

ＳＴＲ$　式

（例）　Ａ$＝ＳＴＲ$１２０　　Ａ$＝"１２０"と同じになります。

Ｂ$＝ＳＴＲ$(１．２＊３)　　Ｂ$＝"３.６"と同じになります。

◉ **ＶＡＬ関数(バリュー)**

ＶＡＬ関数はＳＴＲ$関数とは逆に文字列を数値に変換する関数です。たとえば

Ｂ$＝"1234"のとき

Ａ＝ＶＡＬ　Ｂ$

として実行すればＡには1234という数値が代入されます。この関数の一般形は次のようになります。

ＶＡＬ { “文字” / 文字変数 }

（例）　Ａ＝ＶＡＬ "１２０"　　Ａに120が代入されます。

Ｂ＝ＶＡＬ "３.２＊４="　　Ｂに3.2が代入されます。

VAL関数により数値に変換できる文字列は数値を表わすことのできる数字(０～９.)、符号(＋、－)、指数部を示す記号(Ｅ)で構成されている文字列で、他の文字や記号を含んでいないことが必要です。もし、1つの文字列の中に他の文字や記号が含まれている場合は、それから右の文字列は無視されます。

11-5）アスキーコードの変換（ASC，CHR$）

◉ **アスキー（ASCII）コード**

アルファベット、数字、記号などを計算機が記憶したり、処理したりする場合は、すべて計算機が取り扱いやすい数値に変換します。たとえばアルファベットのAは計算機内では65（10進数）という数値（コード）になっています。（実際には2進数の01000001となっています。）同様にBは66、Cは67というようにコードを決めています。このコードの決めかたに何種類かあって、その1つがアスキーコード（ASCII code）です。

◉ **ASC関数（アスキー）**

この関数は文字や記号、数字などをアスキーコードに変換する関数です。たとえば「Z」という文字のアスキーコードを知りたい場合は

```
A＝ASC"Z"
```

として実行すれば、Aには「Z」のアスキーコードが10進数の90として代入されます。この関数の一般形は次のようになります。

ASC { “文字” / 文字変数 }

なお、文字列が2文字以上指定された場合は、頭の文字のみがアスキーコードに変換されます。

◉ **CHR$関数（キャラクタドル）**

この関数はASC命令とは逆の関数で、10進数のアスキーコードを文字や記号（0～9の数字を含みます。）に変換する関数です。たとえば、アスキーコード「90」の文字を知りたい場合は、

```
A$＝CHR$　90
```

として実行すれば、A$には“Z”が代入されます。CHR$の一般形は次のようになります。

CHR$　式

なお、本機では、文字や記号（これらを総称してキャラクタ：Characterといいます）に対応しているコードは0と32～96までです。ただし0はNUL（空白）です。

（135ページ参照）

11-6）文字列の比較

2つの文字列を比較して、その大小判断をおこなうことができます。(**アスキーコードの大きさによる大小判断**) この判断を用いれば、たとえば文字列をアルファベット順に並べ替えることも容易になります。文字列の比較は、数値の比較に用いました等号、不等号、大小比較の記号を用いることによりおこないます。たとえば

```
"ABC"<"BC"
"AKIRA">"AKIO"
"YAMATO"="YAMATO"
```

(アスキーコードではAが65、Bが66、Cが67……となっています。したがって、文字列の比較では、AはBより小さく、BはCよりも小さくなります。また、数字はアルファベットよりも小さくなります。)

のように書き表わすことができます。いま、文字列をA$とB$とした場合、次の種類の比較ができます。

A$=B$	A$とB$は同じ文字列です。
A$<B$	A$はB$よりも小さい文字列です。
A$>B$	A$はB$よりも大きい文字列です。
A$<=B$	A$はB$よりも小さいか等しい文字列です。
A$>=B$	A$はB$よりも大きいか等しい文字列です。
A$<>B$	A$とB$は同じ文字列ではありません。

これらはIF文の中に使用して大小判断に用いることができます。

=式の中での文字列の長さについて=

文字列の結合(加算)や、大小比較などの式において、文字列の長さ(文字数)の合計は80文字以内でなければなりません。80文字を超えますと文字記憶エリア(計算や比較のため、文字を一時記憶しておくところ)の容量をオーバーしてエラーになります。(ERROR 5)

たとえば、次の式では文字列の長さの合計は9文字になります。

```
X=LEN("ABC"+LEFT$("DEFGHI",2))
```

(この場合、ABCとDEFGHIをすべて文字記憶エリアに入れてから、必要な処理をおこないます。)

12. パスワード(PASS)

作成したプログラムを他の人に知られたくないときや、プログラムを変更されたくないときに、ある言葉や記号をパスワードとし、パスワードが与えられないかぎり、計算機内のプログラムを呼び出すことができないようにすることができます。(プログラムの秘密化)
たとえば、プログラムを書き込んだあと、

PASS"YAMADA" [ENTER]

と操作すれば、計算機内のプログラムがすべて秘密化し、呼び出すことができなくなります。これを呼び出すためには、もう一度同じパスワードを宣言して、秘密化したプログラム(以下秘密プログラムと呼びます)を解除しなければならず、したがって、パスワードを知らなければプログラムを見ることも、変更することもできません。

◉ PASS(パス)

パスワードを設定あるいは解除する命令です。この命令の一般形は次のようになります。

PASS"文字列" [ENTER]
　　└─7文字までの英文字、数字、記号

- パスワードが宣言されていない状態のとき、PASS命令を実行すれば、そのとき計算機内にあるすべてのプログラムに対してパスワードが設定され、秘密プログラムになります。
  秘密プログラムに対してはLIST命令、[↑] や [↓] キーなど、プログラムの呼び出しにかかわる命令や機能およびラインの追加・削除などの機能は働きません。
  また、テープレコーダや他の出力機器に出力することもできません。
- 秘密プログラムを解除する場合はもう一度同じパスワードを宣言します。(パスワードが違っているときはエラー(ERROR 9)になります。)
- 計算機内にプログラムが入っていないときにPASS命令を実行するとエラー(ERROR 1)になり、パスワードは設定されません。
- 秘密プログラムはNEW命令でも消去されず、保護されます。
- 秘密プログラムでないプログラムを秘密プログラムとしてテープに記録することができます。
  秘密プログラムの記録、読み込みについては 77、79 ページをご覧ください。

プリンタに関する命令

これ以降に説明しますプリンタに関する命令は別売のＣＥ-125が接続されているときに有効となる命令です。
ＣＥ-125が接続されていない場合、これらの命令は無視されます。

また、ＣＥ-125のプリンタスイッチ（ＰＲＩＮＴ）がＯＦＦ位置にあるときにプリンタを動作させる命令を実行しますとエラー（ＥＲＲＯＲ　８）になります。（この場合はプリンタスイッチをＯＮ位置にしてから再度実行してください。）

(1)　プリント命令（ＬＰＲＩＮＴ）

計算機の表示命令として、ＰＲＩＮＴ命令がありましたが、これに替わるプリンタの印字命令としてＬＰＲＩＮＴ（ラインプリント）命令があります。
このＬＰＲＩＮＴ命令の一般形はＰＲＩＮＴ命令とほとんど同じ形で指定することができます。

① ＬＰＲＩＮＴ { 式 / “文字” / 文字変数 }

○この形で指定した場合、数値はペーパーの右側に詰めてプリントされ、文字列は左端からプリントされます。
なお、文字列が24桁を越える場合は自動的に改行してプリントします。（ＣＥ-125の１行の印字桁数は24桁です。）

② ＬＰＲＩＮＴ { 式 / “文字” / 文字変数 } , { 式 / “文字” / 文字変数 }

○この形で指定した場合は、１行の印字桁数24桁を左右12桁に分け、，の左右の内容をプリントします。このときも、12桁の範囲内で数値は右ヅメ、文字は左ヅメにします。
なお、表示内容が12桁を越える場合、数値は仮数部の下位桁を切り捨てて12桁以内で、文字は頭から12桁をプリントします。

③ ＬＰＲＩＮＴ { 式 / “文字” / 文字変数 } ; { 式 / “文字” / 文字変数 } ;…; { 式 / “文字” / 文字変数 }

○この形で指定した場合、ペーパーの左端から連続的にプリントされます。プリント内容が24桁を越える場合は自動的に改行され、最大79字までプリントされます。なお、79字目が数値の途中になる場合はエラーになります。

(2) PRINT→LPRINT指定命令

本機はPRINT命令を必要に応じて表示命令と印字命令とに切替えて用いることができます。

たとえば、計算機本体のみで使用しているときは、PRINT命令を表示命令として表示部に計算結果を表示させ、CE-125と接続しているときは印字命令として計算結果などをプリントさせることができます。

◉指定・解除

CE-125が接続されているとき、マニュアルあるいはプログラムで次の命令が実行されれば、PRINT命令はすべてLPRINT命令と同様に働きます。

PRINT=LPRINT

この命令は

PRINT=PRINT

命令で解除することができるほか、RUN命令の実行、[SHIFT] [CA CL] の操作、スライドスイッチで電源をオフ・オンすることにより解除され、PRINT命令は通常の表示命令にもどります。

なお、RUN命令の実行でPRINT=LPRINTの指定は解除されますので、マニュアルでPRINT=LPRINT命令を実行させ、かつ有効に働かせるには、次のような方法を用いてください。

- 命令実行後、GOTO命令でプログラムをスタートさせる。(52ページ参照)
- 命令実行後、定義付けキーによりスタートさせる。(48ページ参照)
- PRINT命令などにより、プログラムがストップしているときに、この命令を実行する。など

=ご注意=

数値を出力する場合、表示部に表示する場合と、プリンタで印字する場合は、数値を表わす方式がかわる場合があります。

たとえば、次のプログラムを実行した場合

```
10:A=1234567,B=-1234567
   8
20:PRINT A,B
30:LPRINT A,B
```

20ラインのＰＲＩＮＴ命令では

```
 1.2E 06-1.2E 07
```

と表示されますが、30ラインのＬＰＲＩＮＴ命令では

```
 1234567.  -12345678.
```

と印字されます。

これは、表示が16桁であるのに対して、印字は１行が24桁になっているためで、表示では固定小数点方式で表わせない場合でも、印字では固定小数点方式で表わせるからです。

なお、ＰＲＩＮＴ命令でも、ＰＲＩＮＴ＝ＬＰＲＩＮＴの指定がなされている場合は、ＬＰＲＩＮＴ命令と同じに動きます。

(3) マニュアル計算でのプリント

マニュアル計算の結果は通常表示部に表示されるのみですが、ＣＥ-125が接続されているときは SHIFT ENTER（P↔NP） と操作して表示部にＰシンボルを表示させておけば、計算の過程や結果がプリントされます。

プリントをさせたくないときは、もう一度 SHIFT ENTER（P↔NP） と操作してＰシンボルを消してください。

- マニュアル計算においてエラーになった場合は、入力されたキー操作のみをプリントし、結果はプリントしません。そして、表示部にエラーの表示をおこないます。
- 入力した内容がＢＡＳＩＣ命令から始まっている場合は、プリントはおこなわれません。

(4) リストのしかた（ＬＬＩＳＴ）

◉ＬＬＩＳＴ命令（ラインリスト）

この命令は計算機に記憶されているプログラムを書き出す命令で、一般形は次のようになります。（**ＰＲＯあるいはＲＵＮモードでのマニュアル操作により実行します。**）

① ＬＬＩＳＴ

プログラムの先頭ラインから、すべてのプログラムを書き出します。

② ＬＬＩＳＴ 式$_1$，式$_2$

式$_1$ の値で示されるラインから 式$_2$ の値で示されるラインまでのプログラムを書き出します。

なお、式$_1$ および 式$_2$ の値がピッタリ一致するラインがない場合は、それぞれの式の値よりも大きくかつ最も近いラインナンバーが指定されます。

ただし、式$_1$ と 式$_2$ で指定されるラインが同じになる場合はエラー（ＥＲＲＯＲ １）になります。

=プリンタに関するご注意=

- プリンタの動作中に紙づまりなどが発生しますと計算機にエラー(ＥＲＲＯＲ　８)が検出されます。
  このときはプリンタにつまった紙を取り除いてから [CL] キーでエラーを解除してください。

- 強度の外来ノイズをうけた場合などに、プリンタが不要なプリントを続け、止まらなくなることなどがありますが、この場合は [ON BRK] キーで動作を止めた後、[CL] キーを押してください。

- ＬＰＲＩＮＴ命令の実行などによりプリントがおこなわれているときに、INPUT、INKEY＄、ＰＲＩＮＴ命令が実行された場合、プリント中でも計算機のキー入力ができるようになりますが、このとき [CL] キーを押しますと、プリンタは途中でプリントを中止しますのでご注意ください。
  [CL] キーはプリントが終了してから押してください。

- プリンタの電池電圧が低くなりますと、プリンタは動作しなくなります。この場合は電池交換あるいは充電をおこなってください。
- プリンタの紙送りキー([↑])はプリントをおこなっているときは動作しません。

テープレコーダーに関する命令

ここではテープレコーダーを使用して、テープにプログラムやデータを記録したり、計算機に読み込んだり(転送したり)、あるいは記録されている内容を照合したりする命令について説明します。

(1) プログラムの記録命令(ＣＳＡＶＥ)

◉ **ＣＳＡＶＥ命令(カセット・セイブ)**

プログラムをテープに記録する(セイブする)命令で、次のような形で指定します。

① ＣＳＡＶＥ〝ファイル名〝 ENTER

○プログラムの記録

プログラムをテープに記録する場合はＰＲＯモードかＲＵＮモードでＣＳＡＶＥ命令を実行します。

(例) ＰＲＯモードかＲＵＮモードを指定
ＣＳＡＶＥ〝ＰＲＯＧ．－１〝 ENTER

ファイル名として「ＰＲＯＧ.－1」をテープに記録したのち、計算機内のプログラムをすべて記録します。

注) 同じテープの同じ面(Ａ面あるいはＢ面)に、同じファイル名で内容の違うプログラムを記録することは避けてください。同じファイル名のものが2つ以上ありますと、テープから読み込む(転送する)ときにまちがった内容を読み込む恐れがあります。

記録をおこなうとき、テープレコーダーのテープカウンターの数字を紙などに控えておけば、転送などをおこなうとき、記録した部分を簡単に探すことができます。

○ファイル名

ファイル名はプログラムなどにつける名前(見出し)のことで、ファイル名をつけてプログラムなどを記録しておけば、あとで転送などをおこなうとき、計算機によりファイル名を探し出して転送などをおこなうことができますので便利です。なお、ファイル名を省略することもできます。

ＣＳＡＶＥ命令は、計算機内のプログラムが秘密プログラムのときは働きません。
したがって秘密化されているプログラムをテープに記録するときはＰＡＳＳ命令で解除してください。
なお、テープに記録するプログラムを秘密化するときは次の方法でおこなってください。

○記録するプログラムの秘密化

テープに記録するプログラムを秘密化するときは次の形で記録してください。

② CSAVE〝ファイル名〟,〝パスワード〟 ENTER
あるいはCSAVE,〝パスワード〟 ENTER

この形でCSAVE命令を実行すれば、プログラムはパスワードつきのプログラムとして記録され、後述のCLOAD命令などで計算機に読み込まれたときは、秘密プログラムとなります。
したがって、解除するにはPASS命令でパスワードを宣言しなければなりません。

●パスワードは最大7文字までの英文字、数字、記号などを用いることができます。

(2) プログラムの転送命令(CLOAD, MERGE, CHAIN)

転送するときはテープを巻きもどし、あるいは送って、転送する内容が記録されている部分の直前までテープを進めてください。

◉ CLOAD命令(カセット・ロード)

テープに記録されているプログラムを計算機に転送するための命令で、次のような形で指定します。なお、この命令は**マニュアル操作でのみ実行**できます。

① CLOAD“ファイル名” ENTER
あるいは CLOAD ENTER

○プログラムの転送

テープに記録されているプログラムを計算機に転送する場合はPROモードかRUNモードでCLOAD命令を実行します。

(例) PROモードかRUNモードを指定
CLOAD〝PROG.−1〟 ENTER

ファイル名が「PROG.−1」のプログラムを検索して、これを計算機に転送します。

ファイル名をつけずにCLOAD ENTER と操作した場合はテープレコーダーが動き始めて最初に記録されているプログラムを計算機に転送します。

●転送がおこなわれているときは、表示部の右端の桁に＊マークが表示されます。そして、転送が終れば、＊マークは消えます。
ファイル名を検索しているときは、まだ転送がおこなわれていませんので＊マークは表示されません。
これは、のちほど説明しますMERGE、CHAIN、CLOAD?およびINPUT＃命令でも同じです。

注1）指定されたファイル名が検索できなかったときは、テープが回り終っても計算機はファイル名を検索し続けます。この場合は ON BRK キーを押して検索を止めてください。これは、のちほど説明しますMERGE、CHAIN、CLOAD?、およびINPUT#命令でも同じです。

注2）CLOAD命令およびのちほど説明のCHAIN命令を実行中にエラーが発生しますと、計算機内のプログラムは無効になります。

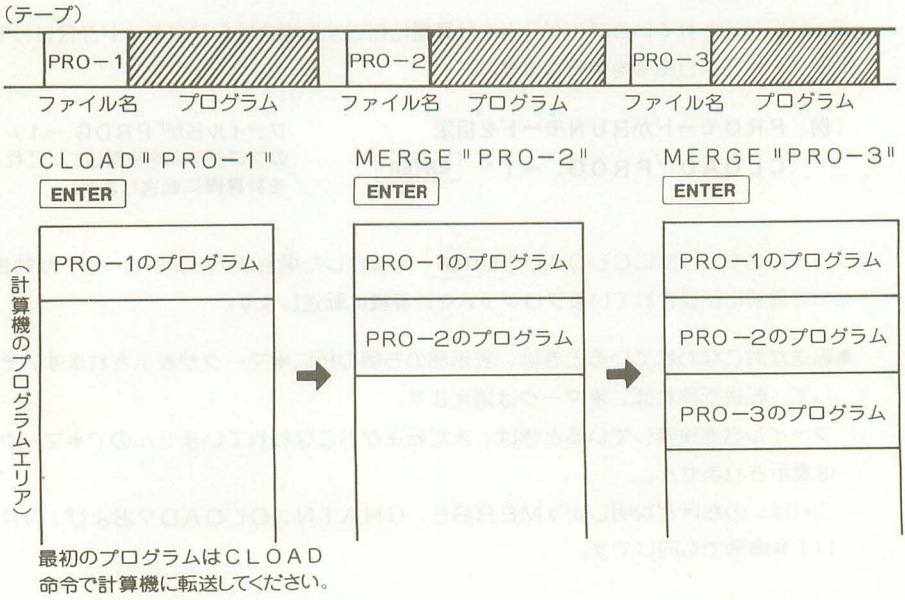
◉ **MERGE命令（マージ）**

MERGE命令はテープに記録されているプログラムを計算機に転送する命令で、CLOAD命令と同じ形で指定し、実行します。（マニュアル操作でのみ実行できます。）

① MERGE “ファイル名” ENTER
あるいは MERGE ENTER

CLOAD命令とMERGE命令の違いは、CLOAD命令が前に計算機に記憶されていたプログラムをすべて消してからテープに記録されているプログラムを読み込んでくるのに対して、MERGE命令は前のプログラムを保持したまま、それらの後に、テープに記録されている内容を追加していく形で読み込みます。したがって、全然別のプログラムを計算機に同時に記憶させておくことができます。

（例）テープにPRO−1〜PRO−3のファイル名を持つプログラムが記憶されているとき、PRO−1をCLOAD命令で、PRO−2、PRO−3をMERGE命令で計算機に転送した場合の記憶の状態は次のようになります。



MERGE命令により読み込まれたプログラムは前例のように記憶されますが、このとき、それぞれのプログラムのラインナンバーにより、プログラムの扱いが次のようになります。

- ●MERGE命令により読み込まれたプログラムの最初のラインナンバーが、すぐ前に入っているプログラムの最終ラインのナンバーよりも大きい場合、2つのプログラムは1つのプログラムと見なされます。
- ●MERGE命令により読み込まれたプログラムの最初のラインナンバーが、すぐ前に入っているプログラムの最終ラインのナンバーよりも小さい場合はそれぞれ独立したプログラムと見なされます。

たとえば、先の例でPRO－1、PRO－2、PRO－3のプログラムのラインナンバーがそれぞれ10～200、50～150、160～300の場合は、PRO－1とPRO－2はそれぞれ独立したプログラムとなりますが、PRO－2とPRO－3は50～300ラインの1本のプログラムと見なされます。

○MERGE命令によりプログラムを読み込ませますと、計算機内に同じラインナンバーを持つプログラムが2つ、3つ…と存在するようになります。
この場合、プログラムを実行するRUN命令やGOTO命令は最初のプログラムに対してのみ有効となり、2番目、3番目…のプログラムを実行する方法がなくなりますので、これらのプログラムは事前に“定義付けプログラム”として、[A]、[S]、[D]…などのキーに定義付けしておく必要があります。
（定義付けプログラムについては48ページを参照ください。）

- ●MERGE命令実行後のプログラム編集は最後に転送し、記憶されているプログラムに対してのみおこなうことができます。

○秘密プログラムのMERGE
パスワードのついたプログラム(秘密プログラム)をMERGE命令で読み込むときは、計算機内のプログラムが秘密化されているかいないかによって、プログラムの扱いが次のようにかわります。

- ●秘密化されているとき
  秘密プログラムを読み込むことはできません。

- ●秘密化されていないとき
  秘密プログラムをMERGE命令で読み込めば、計算機内のプログラムがすべて秘密化されます。

なお、計算機内のプログラムが秘密化されているとき、パスワードのないプログラムをMERGE命令で読み込んだ場合はそのプログラムも秘密化されます。

◉ CHAIN命令(チェイン)

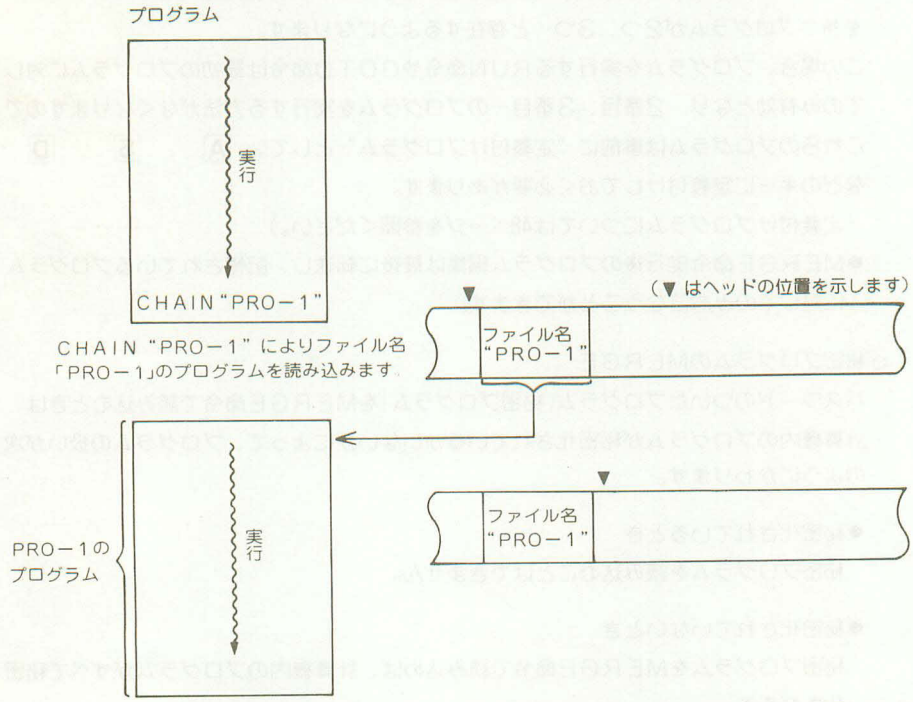
CHAIN命令は**プログラムの中で使用する命令**で、テープに記録されているプログラムを計算機に読み込んで、そのプログラムを実行していく命令です。(プログラムでのみ実行できます。)この命令を用いますと、計算機のプログラムエリア内に収まり切れないような長いプログラムでも、何分割かしてテープに記録しておき、次々と読み込みながら実行を続けていくことができます。

CHAIN命令の一般形は次のようになります。

① CHAIN "ファイル名"
CHAIN "ファイル名", { 式 / "ラベル" }

指定されたファイル名のついたプログラムを読み込み、そのプログラムの先頭ライン、あるいは式の値(あるいはラベル)で指定されたラインから実行します。なお、ファイル名を省略した場合は、テープが回り出して最初のプログラムを読み込みます。

(例) CHAIN命令により、次のようにプログラムを読み込みます。



このようにプログラムの最後にCHAIN文を入れておくと、プログラム実行の最後に自動的にテープから次のプログラムを読み込んで実行を続けていくことが可能になります。

(3) 照合命令（CLOAD?）

◉ **CLOAD?命令(カセット・ロード?)**

計算機に記憶させているプログラムと、テープに記憶されている内容との照合をおこなう命令で、次のような形で指定します。なお、この命令は**マニュアル操作による実行のみ可能**です。

① CLOAD?“ファイル名” ENTER
あるいは CLOAD? ENTER

この命令により、もし不一致が生じたときは、エラー(ERROR 8)になります。
不一致がなければプロンプト表示になります。

○プログラムの照合

計算機に記憶されているプログラムとテープに記録されているプログラムを照合する場合はPROモードあるいはRUNモードでCLOAD?命令を実行します。

（例）PROモードかRUNモードを指定

CLOAD?〝PRO−1〝 ENTER

ファイル名が「PRO−1」のプログラムを検索し、その内容と、計算機のプログラムとを照合します。

●プログラムの照合がおこなわれているときは表示部の右端の桁に＊マークが表示されます。
照合が終れば＊マークが消えます。

(4) データの記録命令(ＰＲＩＮＴ＃)

◉ **ＰＲＩＮＴ＃命令(プリント・クロスハッチ)**

計算機内に記憶されているデータをテープに記録する命令で、次のような形で指定します。(マニュアル操作、プログラムとも実行できる命令です。)

① ＰＲＩＮＴ＃“ファイル名”
　あるいはＰＲＩＮＴ＃

この形では固定変数及びA(　)、Ａ$(　)の形の変数の内容をすべてテープに記録します。

注) A～ZとA(１)～A(26)とは同じ内容の変数です。

② ＰＲＩＮＴ＃“ファイル名”；変数名，変数名，…
　あるいはＰＲＩＮＴ＃変数名，変数名，…

この形では、変数名がA(　)の形の場合は指定された変数も含めて、添字がそれより大きいA(　)変数の内容をすべてテープに記録します。(これはＡ、Ｂ、Ｃ…の形で指定した場合も同様です。)

たとえばA(５)あるいはＥを指定した場合はA(５)～A(26)すなわちＥ～Ｚの変数の内容と、A(27)以降の変数が使用(確保)されているときはその変数の内容をすべてテープに記録します。

変数名が配列変数の場合は、その配列名を持つ変数の内容がすべて記録されます。

○変数名

変数名は、Ａ、Ｂ、Ｃ…あるいはA(１)、A(２)、A(３)…の形で指定します。

また、A(　)以外の配列変数の場合は(　)の中に＊を指定します。この形で、同じ名前の配列変数すべてを同時に指定します。

A(　)以外の配列変数は数値変数、文字変数とも、この形でのみ記録することが可能です。

(例) ＰＲＩＮＴ＃Ｃ，Ｂ(＊)	変数Ｃ～Ｚ、A(27)～およびB(　)の変数(B(0)、B(1)、B(2)…)の内容をファイル名なしでテープに記録します。
ＰＲＩＮＴ＃〝ＤＡＴＡ－１〟；A(1)，Ｃ$(＊)	ファイル名「ＤＡＴＡ－１」をつけてA(1)～A(26)(Ａ～Ｚ)、A(27)～およびＣ$(　)の変数(Ｃ$(0)、Ｃ$(1)…)をテープに記録します。

注) A(27)以上の変数や、配列変数はＰＲＩＮＴ＃命令を実行する前に、プログラム・データエリアに確保されていなければなりません。

もし、ＰＲＩＮＴ＃に続いて指定された変数が確保されていない場合にはエラー(ＥＲＲＯＲ　３)になります。

(5) データの転送命令(INPUT＃)

◉ INPUT＃命令(インプット・クロスハッチ)

テープに記録されているデータを指定された変数に転送する命令で、次のような形で指定します。(マニュアル操作、プログラムとも実行できる命令です。)

① INPUT＃“ファイル名”
あるいはINPUT＃

この形では、テープに記録されているデータを固定変数およびA( )の形の変数(A(27)～)に順番に転送していきます。
転送はテープに記録されているデータがなくなるまで、または計算機内のメモリーがいっぱいになるまでおこなわれます。

② INPUT＃“ファイル名”；変数名，変数名，…
あるいはINPUT＃変数名，変数名，…

この形では、変数名がA( )の形の場合は指定された変数も含めて、添字がそれより大きいA( )変数に、テープに記録されているデータを順番に転送していきます。
(固定変数の場合も同様です。)
また、変数名がA( )以外の配列変数の場合は配列名(＊)と指定することにより、テープに記録されているデータを指定した配列変数に順番に転送していきます。
なお、A( )以外の配列変数は、INPUT＃命令を実行する前にDIM命令により変数名と大きさを定義しなければなりません。

(例) INPUT＃ A(3)

テープが回りはじめて最初に記録されているデータを変数A(3)～A(26)(すなわちC～Z)、A(27)～に転送していきます。

```
(例) 50 DIM B(5)
     60 INPUT＃"DATA－1";B(＊)
     ⋮
```

ファイル名が「DATA－1」のデータを検索して、変数B(0)～B(5)に転送していきます。

注) テープに記録されたデータを変数に転送する場合、データと変数とはその形(数値変数か文字変数かの形)、大きさ、長さが一致していなければなりません。
もし、大きさ、長さが一致しなければエラー(ERROR 8)になります。
ただし、数値データを文字変数へ転送したり、文字データを数値変数へ転送したりしますと、エラーにならず、誤ったデータを転送することがありますので、データと変数の形は必らず一致させてください。

注) 固定変数及びA( )の形の変数へのデータの転送は、テープに記録されているデータがなくなるか、または計算機内のメモリーがいっぱいになれば終了します。

オプション機器のご案内

PC-1245は、次の別売周辺機器が使用できます。

● CE-124

カセットインターフェイス

CE-152などのテープレコーダーとPC-1245をつなぐためのインターフェイスです。

● CE-152

ポケットコンピュータ専用カセットテープレコーダー

CE-124あるいはCE-126Pを介してプログラムやデータをテープに記録、転送ができます。

● CE-125

プリンタ/マイクロカセットレコーダー

サーマルプリンタとマイクロカセットレコーダーを一体化し、コンパクトにまとめた、プリンタ/マイクロカセットレコーダーです。CE-125をお使いいただくことにより、プログラムや計算結果を書き出したり、プログラムやデータをテープに記録したり、その内容をテープから読み込んだりすることができます。

注）CE-125に付属のマイクロカセットテープに収録されていますプログラムは、そのままではご利用いただけません。

（108ページを参照ください。）

● CE-126P

プリンタ/カセットインターフェイス

カセットインターフェイスを内蔵したサーマルプリンタです。CE-126Pをお使いいただくことにより、プログラムを書き出したり計算結果などを書き出したりすることができます。また、CE-126Pを介してプログラムやデータをテープに記録、転送できます。

＝ご注意＝

CE-126Pをお使いいただく場合、キャラクターコード39(&27)に割り当てられている内容が計算機(表示)とCE-126P(印字)では異っていますのでご注意ください。

（例）次のプログラムを実行した場合

```
10  B$=CHR$  39
20  PRINT  B$
30  LRRINT  B$
RUNモード
```

RUN [ENTER]　␣←20ラインで、計算機の持っているキャラクタコード39の内容を表示。

[ENTER]　←30ラインで、CE-126Pの持っているキャラクタコード39の内容(スペース)を印字。(この例では1行分紙送りをします。)

計算機とテープレコーダーの操作方法

1．CE-125を使用される場合

プログラムなどをテープに記録する、あるいはテープから転送する場合の計算機とテープレコーダーの操作方法について説明します。

(1) 準　備

◉ 計算機とCE-125を接続してください。

◉ テープレコーダーに関する注意

- テープレコーダーのヘッドなどが汚れていますと、ひずみの増加、レベルの低下などがおこりますので、ヘッドの清掃をおこなってください。
- テープは一般に市販のマイクロカセットテープ（MC-60ノーマルタイプ）をご使用ください。ただし、極端に周波数特性の悪いもの、また、傷や折目があるものは使用しないでください。
- 一度使用したテープに新しくプログラムなどを書き込む場合は、書き始める前を少し消去してから記録命令を実行してください。

(2) テープへの記録方法

例としてプログラムを記録する方法について説明します。（マニュアル操作の場合）

〈プログラム〉

```
10  INPUT"HANKEI␣R=";A
20  S=π*A*A
30  PRINT"MENSEKI␣S=";S
40  END
```

手順	操　　作	備　　考
(1)	CE-125のリモートスイッチをOFF側にしてください。	
(2)	テープをテープレコーダーにセットしてください。	テープの中の記録されていない部分をさがし、記録する位置をきめてください。
(3)	記録する内容を計算機に入れてください。たとえば、前ページのプログラムを計算機に書き込んでください。	
(4)	CE-125のリモートスイッチを、ON側にしてください。	
(5)	テープレコーダーの記録ボタン(●)を押し込んで、録音状態にしてください。	
(6)	記録命令を実行します。 プログラム：CSAVE“ファイル名” ENTER データ：PRINT#“ファイル名”；変数名，変数名… ENTER （例）CSAVE〝PR−1〟 ENTER	記録命令が実行されますと、まず最初に約8秒間無信号区間が記録(ピー音が記録)され、そのあとにファイル名とプログラムなどが記録されます。(プログラムなどの記録中はピピピピ……あるいはビビビビ…というような音が出ます。)
(7)	記録が終了しますと、プロンプト表示になり、テープが止まります。 (巻きもどし、早送りなどをおこなうときは、CE-125のリモートスイッチをOFF側にして、おこなってください)	照合については、87ページを参照ください。

●プログラムの実行において、データの記録をおこなう場合(PRINT#命令)には、プログラム実行前にテープレコーダーの操作をしておいてください。

記録をおこなうとき、記録を始めたときのテープカウンタの数値を紙などに控えておけば、記録位置や、記録されていない位置を探し出すときに便利です。

(3) 計算機とテープの内容の照合

計算機に記憶されているプログラムと、テープに記録されているプログラムとの照合を取る場合(マニュアル操作でのみ実行可能)には次のようにおこないます。

手順	操　　作	備　　考
(1)	CE-125のリモートスイッチをOFF側にしてください。	
(2)	照合する内容が記録されているテープをテープレコーダーにセットしてください。 そして、テープカウンターなどを見ながら照合する内容の記録されている位置の近く(少し前)までテープを送ってください。	
(3)	CE-125のリモートスイッチをON側にしてください。	
(4)	テープレコーダーの再生ボタン(◀)を押し込んで再生状態にしてください。	
(5)	照合命令を実行します。 CLOAD?“ファイル名” ENTER (例) CLOAD?"PR−1" 照合がおこなわれているときは表示部の右端の桁に＊マークが表示されます。 両方の内容が一致している場合は実行を終了しプロンプト表示になります。 もし、エラー表示(ERROR 8)になった場合は、もう一度最初から照合をおこなってください。 それでもなお、エラーになる場合は、もう一度“記録”からおこなってください。	ファイル名は記録をしたファイル名を入れてください。ファイル名を省略した場合はテープが回り出して最初の記録内容と、計算機内の内容との照合をおこないます。 ファイル名にまちがいがないか、よく確認しながら操作してください。

＝ご注意＝

テープレコーダーの再生ボタン(◀)が押されていて、転送命令や照合命令が実行されているときに、早送りボタン(◀◀)や巻きもどしボタン(▶▶)を押さないでください。
早送りや巻きもどしをおこないますと、転送や照合が正常におこなわれません。

(4) テープからの転送

テープに記録されているプログラムあるいはデータを計算機に転送する（テープより読み出す）場合は次のようにおこなってください。(マニュアル操作の場合)

手順	操　　作	備　　考
(1)	CE-125のリモートスイッチをOFF側にしてください。	
(2)	転送する内容が記録されているテープをテープレコーダーにセットしてください。 そして、テープカウンターなどを見ながら転送する内容の記録されている位置の近く(少し前)までテープを送ってください。	
(3)	CE-125のリモートスイッチをON側にしてください。	
(4)	テープレコーダーの再生ボタン（◀）を押し込んで再生状態にしてください。	
(5)	転送命令を実行します。 プログラム：CLOAD “ファイル名” ENTER データ：INPUT#“ファイル名”；変数名，変数名，… ENTER （例） CLOAD "PR−1" ENTER	ファイル名は記録をしたファイル名を入れてください。 ファイル名を省略した場合はテープが回り出して最初に記録されている内容が転送されます。 転送がおこなわれているときは表示部の右端の桁に＊マークが表示されます。 転送が終了すればプロンプト表示になります。
(6)	もし、途中でエラー（ERROR 8）になった場合は、もう一度最初から転送をおこなってください。 それでもなお、エラーになる場合は、もう一度“記録”からおこなってください。	ファイル名にまちがいがないか、よく確認しながら操作してください。

●プログラムの実行において、転送をおこなう場合(CHAIN命令、INPUT#命令)には、プログラム実行前にテープレコーダーの操作をしておいてください。

2．CE-124またはCE-126Pを使用される場合

注）テープレコーダーは別売のポケットコンピュータ専用カセットテープレコーダーCE-152を使用されているものとして説明いたします。

(1) 準　備

◉ 計算機、CE-124またはCE-126P、テープレコーダー（CE-152）を接続してください。（接続方法はCE-124、CE-126P及びCE-152の取扱説明書を参照ください。）

◉ テープレコーダーに関する注意

- 転送あるいは照合をおこなうときは、記録をおこなったテープレコーダーを使用してください。
  記録したテープレコーダーと転送あるいは照合するテープレコーダーが異なる場合は、転送や照合ができないことがあります。
- テープレコーダーのヘッドなどが汚れていますと、ひずみの増加、レベルの低下などがおこりますので、ヘッドの清掃をおこなってください。
- テープは一般に市販のものをご使用ください。ただし、極端に周波数特性の悪いもの、また、傷や折目があるものは使用しないでください。
- 一度使用したテープに新しくプログラムなどを書き込む場合は、書き始める前を少し消去してから記録命令を実行してください。

(2) テープへの記録方法

例としてプログラムを記録する方法について説明します。（マニュアル操作の場合）

〈プログラム〉

```
10  INPUT "HANKEI␣R=" ; A
20  S=π*A*A
30  PRINT "MENSEKI␣S=" ; S
40  END
```

手順	CE-124を用いる場合	CE-126Pを用いる場合
(1)	テープをテープレコーダーにセットしてください。	CE-126Pのリモートスイッチを OFF側にして、テープをテープレコーダーにセットしてください。
	●テープの中の記録されていない部分をさがし、記録する位置をきめてください。（テープの始めの無録音部分はさけてください。）	
(2)	記録する内容を計算機に入れてください。たとえば、前ページのプログラムを計算機に書き込んでください。	記録する内容を計算機に入れてください。たとえば、前ページのプログラムを計算機に書き込んでください。
(3)	テープレコーダーの一時停止ボタン(⏸)を押し込んで、一時停止状態にしたうえで、記録ボタン(●)を押して、録音可能な状態にしてください。	CE-126Pのリモートスイッチを、ON側にしてください。 そしてテープレコーダーの記録ボタン(●)を押し込んで、録音状態にしてください。
(4)	記録命令を入力してください。 プログラム：CSAVE〃ファイル名〃 データ：PRINT#〃ファイル名〃; 変数名，変数名…… （例）PROモードかRUNモードを指定 CSAVE〃PR-1〃	記録命令を入力してください。 プログラム：CSAVE〃ファイル名〃 データ：PRINT#〃ファイル名〃; 変数名，変数名…… （例）PROモードかRUNモードを指定 CSAVE〃PR-1〃
(5)	記録命令を実行します。 一時停止ボタン(⏸)を押して一時停止状態を解除し、ENTER キーを押してください。	記録命令を実行します。 ENTER キーを押してください。
	●記録命令が実行されますと、まず最初に約8秒間無信号区間が記録（ピー音が記録）され、そのあとにファイル名とプログラムなどが記録されます。(プログラムなどの記録中はピピピピ…あるいはビビビビ…というような音が出ます。)	
(6)	記録が終了しますと、プロンプト表示になります。 テープを巻きもどして、“照合”をおこなってください。	記録が終了しますと、プロンプト表示になり、テープが止まります。 リモートスイッチをOFF側にして、テープを巻きもどし、“照合”をおこなってください。

●プログラムの実行において、データの記録(PRINT＃命令の実行)をおこなうとき、CE-126Pをご使用の場合は、プログラム実行前にテープレコーダーの操作をしておいてください。

CE-124をご使用の場合は、PRINT＃命令の前にSTOP命令を入れ、プログラムの実行を停止させるようにしてください。

(例)

```
    ⋮
150：STOP：PRINT＃"ABC";F,B(＊)
    ⋮
```

そして、STOP命令によりプログラムが停止しているときに、テープレコーダーの操作をおこなってください。

次に、CONTとキー入力してからテープをスタートさせ、テープが回りはじめてから ENTER キーを押せばプログラムが再スタートして、STOP命令に続くPRINT＃命令が実行されて、データの記録がおこなわれます。

●プログラムを停止させる方法としては、PRINT命令を用いる方法もあります。

(例)

```
    ⋮
140：PRINT"TAPE START"
150：PRINT＃"ABC";F,B (＊)
    ⋮
```

ただしこの場合、プログラムの中にWAIT命令が使用されているときは、その指定を解除するなどの配慮が必要になります。

記録をおこなうとき、記録を始めたときのテープカウンタの数値を紙などに控えておけば、記録位置や、記録されていない位置を探し出すときに便利です。

(3) 計算機とテープの内容の照合

計算機に記憶されているプログラムと、テープに記録されているプログラムとの照合を取る場合(マニュアル操作でのみ実行可能)には次のようにおこないます。

<table>
<tr><th>手順</th><th>CE-124を用いる場合</th><th>CE-126Pを用いる場合</th></tr>
<tr><td>(1)</td><td>照合する内容が記録されているテープをテープレコーダーにセットしてください。</td><td>CE-126Pのリモートスイッチを OFF側にして、照合する内容が記録されているテープをテープレコーダーにセットしてください。</td></tr>
<tr><td></td><td colspan="2">●テープカウンターなどを見ながら照合する内容の記録されている位置が、以後の回転位置にあるようにしてください。</td></tr>
<tr><td>(2)</td><td>音量調整つまみ、音質調整つまみを矢印方向いっぱいに回しておいてください。
H◀TONE▶L　VOLUME◀MAX
 </td><td>音量調整つまみ、音質調整つまみを矢印方向いっぱいに回しておいてください。
H◀TONE▶L　VOLUME◀MAX
 </td></tr>
<tr><td>(3)</td><td>テープレコーダーの一時停止ボタン(⏸)を押し込んで、テープレコーダーを一時停止の状態にしてください。
続いて再生ボタン(▶)を押して、再生状態にしてください。</td><td>CE-126Pのリモートスイッチを ON側にしてください。
そして、テープレコーダーの再生ボタン(▶)を押し込んでください。</td></tr>
<tr><td>(4)</td><td>照合命令を入力してください。
CLOAD?“ファイル名”
(例) PROモードかRUNモードを指定
CLOAD?〃PR-1〃</td><td>照合命令を入力してください。
CLOAD?“ファイル名”
(例) PROモードかRUNモードを指定
CLOAD?〃PR-1〃</td></tr>
<tr><td>(5)</td><td>照合命令を実行します。
ENTER キーを押し、続いてテープレコーダーの一時停止ボタン(⏸)により一時停止状態を解除してください。</td><td>照合命令を実行します。
ENTER キーを押してください。</td></tr>
<tr><td></td><td colspan="2">●照合がおこなわれているときは、表示部の右端桁に＊マークが表示されます。</td></tr>
<tr><td></td><td colspan="2">両方の内容が一致している場合は実行を終了しプロンプト表示になります。
もし、エラー表示(ERROR 8)になった場合は、もう一度最初から照合をおこなってください。
それでもなお、エラーになる場合は、もう一度“記録”からおこなってください。</td></tr>
</table>

(4) テープからの転送

テープに記録されているプログラムあるいはデータを計算機に転送する(テープより読み出す)場合は次のようにおこなってください。(マニュアル操作の場合)

手順	CE-124を用いる場合	CE-126Pを用いる場合
(1)	転送する内容が記録されているテープをテープレコーダーにセットしてください。	CE-126Pのリモートスイッチを OFF側にして、転送する内容が記録されているテープを テープレコーダーにセットしてください。
	●テープカウンターなどを見ながら転送する内容の記録されている位置が、以後の回転位置にあるようにしてください。	
(2)	音量調整つまみ、音質調整つまみを矢印方向いっぱいに回しておいてください。 H◀TONE▶L　VOLUME◀MAX 	音量調整つまみ、音質調整つまみを矢印方向いっぱいに回しておいてください。 H◀TONE▶L　VOLUME◀MAX
(3)	テープレコーダーの一時停止ボタン(⏸)を押し込んで、テープレコーダーを一時停止の状態にしてください。 続いて再生ボタン(▶)を押して、再生状態にしてください。	CE-126Pのリモートスイッチを ON側にしてください。 そして、テープレコーダーの再生ボタン(▶)を押し込んでください。
(4)	転送命令を入力してください。 プログラム：CLOAD"ファイル名" データ　：INPUT#"ファイル名" ;変数名，変数名… (例) PROモードかRUNモードを指定 CLOAD"PR-1"	転送命令を入力してください。 プログラム：CLOAD"ファイル名" データ　：INPUT#"ファイル名" ;変数名，変数名… (例) PROモードかRUNモードを指定 CLOAD"PR-1"
(5)	転送命令を実行します。 ENTER キーを押し、続いてテープレコーダーの一時停止ボタン(⏸)により一時停止状態を解除してください。 ●転送がおこなわれているときは、表示部の右端の桁に＊マークが表示されます。	転送命令を実行します。 ENTER キーを押してください。

	転送が終了しますと、プロンプト表示になります。 もし途中でエラー（ＥＲＲＯＲ８）になった場合は、もう一度最初から転送をおこなってください。 それでもなお、エラーになる場合は、もう一度“記録”からおこなってください。

●プログラムの実行において、転送をおこなうとき（ＣＨＡＩＮ命令、ＩＮＰＵＴ＃命令）、ＣＥ-126Ｐをご使用の場合は、プログラム実行前にテープレコーダーの操作をしておいてください。
ＣＥ-124をご使用の場合は、プログラムにおいてＣＨＡＩＮ命令、ＩＮＰＵＴ＃命令の直前にＳＴＯＰ命令を入れてプログラムの実行を停止させ、テープレコーダーの操作をおこなってください。

キーの機能

次に主なキーの機能を説明します。

キー	機能
ON BRK	○オートパワーオフ機能により切れている電源を入れるときに押します。 ○プログラム実行中、プログラムを一時停止状態（ＢＲＥＡＫ状態……ブレーク）にします。 ○ＣＬＯＡＤ命令などの実行中は、その実行を中止します。
SHIFT	○π、∧、？のように、茶色でパネル上に記されている命令（機能）を指定します。 （例） SHIFT ? U → ？が入力されます。 SHIFT INPUT A → INPUT が入力されます。
CL	○入力中の内容や表示をクリアするときに押します。（エラー状態を解除します。）
SHIFT CA	○計算機の状態を解除します。（クリアオール） ●プログラムの実行が一時停止状態にあるとき、実行を中止させます。 ●ＷＡＩＴ命令による時間指定を解除します。 ●表示フォーマット指定（ＵＳＩＮＧ指定）を解除します。 ●トレースモードの解除（ＴＲＯＦＦ状態にする） ●PRINT＝LPRINT指定の解除 ●エラー状態の解除 その他
0 ～ 9	○数値を指定します。
•	○小数点を指定します。 ○命令入力の際の省略形の指定をおこないます。 ○ＵＳＩＮＧ命令内において、数値データの表示に小数点以下の表示をおこなう指定をします。
SHIFT EXP	○指数部入力を指定します。
／	○除算命令を指定します。
＊	○乗算命令を指定します。 ○ＤＩＭ命令やＰＲＩＮＴ＃、ＩＮＰＵＴ＃命令の中で配列変数を指定するときに使用します。

[+]	○数値の符号（正符号）を指定します。（一般に省略可） ○加算命令を指定します。
[−]	○数値の符号（負符号）を指定します。 ○減算命令を指定します。
[SHIFT] [∧]	○べき乗命令を指定します。 ○ＵＳＩＮＧ命令内において、数値データの表示形式を浮動小数点方式（指数方式）に指定します。
[SHIFT] [<] [SHIFT] [>]	○ＩＦ文の条件式などに使用します。 ○論理式を指定します。
[DEF]	○Ａ、Ｓ、Ｄ、…などのラベルで定義されたプログラムの実行を開始するとき、そのラベルに対するキーを押す前に押します。（Definableキー指定）
[A] ～ [Z]	○アルファベットキーで、各命令や、変数名などを入力します。 ○Ａ、Ｓ、Ｄ…などは定義付けキーとして使用できます。
[SPC]	○プログラム入力あるいは文字列入力時にスペースを設けるためのキーです。 〃　〃内及びＲＥＭ文内以外のスペースはプログラム、演算などの実行時にはないものとみなされます。 ○定義付けキーとして使用できます。
[=]	○代入文において、右辺の内容を左辺で指定される変数に代入することを指定します。 ○ＩＦ文の条件式などに使用します。 ○定義付けキーとして使用できます。
[SHIFT] [INPUT] [SHIFT] [IF] 〜 [SHIFT] [CLOAD]	○それぞれのキーに定義されているＢＡＳＩＣ命令を入力します。

SHIFT !	○それぞれのキャラクターシンボルを入力します。
SHIFT "	"…文字列の指定をおこないます。 ラベルの指定をおこないます。
SHIFT #	#…ＵＳＩＮＧ命令内において、数値データの表示フォーマットを定義します。 ＰＲＩＮＴ#命令およびＩＮＰＵＴ#命令を指定するときに使用します。
SHIFT $	$…文字変数を指定します。
SHIFT %	&…ＵＳＩＮＧ命令内において、文字列の表示フォーマットを定義します。 16進数値を示します。
SHIFT &	
SHIFT ?	?…ＣＬＯＡＤ?命令を指定するときに使用します。
SHIFT :	:…１ラインに複数のステートメントを定義するときの区切りを指定します。
SHIFT ,	,…式と式の区切りや変数の区切り、コメントの区切りなどを指定します。
SHIFT ;	;…式や変数の区切り、コメントの区切り、命令文と変数の区切りなどを指定します。
SHIFT @	@！%¥…"　　"の中で文字列として使用します。
SHIFT ¥	
SHIFT (	○カッコの指定をおこないます。
SHIFT)	
►	○カーソルの右シフトをおこないます。 ○プレイバック命令の実行をおこないます。 ○プログラム内容が表示されているとき、カーソルが表示されていない場合にカーソルを呼び出します。 ○マニュアル操作時のエラーを解除します。
◄	○カーソルの左シフトをおこないます。 ○そのほか ► キーと同じ。
SHIFT INS	○カーソルが示す位置に、１ステップ容量の挿入スペース（ただし□表示）を設けます。
SHIFT DEL	○カーソルが示す位置の内容を削除します。
SHIFT π	○π（パイ）を指定します。
SHIFT √	○開平計算の命令を指定します。

ENTER	○プログラムのラインの終了命令を指定します。 ○プログラム、の書き込みをおこないます。 ○マニュアル計算の実行あるいはＢＡＳＩＣ命令のマニュアル操作による実行を命令します。 ○ＩＮＰＵＴ命令やＰＲＩＮＴ命令で、一時停止しているプログラムの再スタートを命令します。
SHIFT P↔NP	○ＣＥ-125が接続されているとき、マニュアル計算のプリント↔ノンプリントモードを指定します。プリントモードでは表示部にＰシンボルが表示され、計算の過程や結果がプリントされます。

↓ 、↑ キーについては、モードの指定および計算機の状態により、次のように働きます。

モード	状　態	↓	↑
RUN	プログラム実行中		
	プログラムの一時停止中 INPUT命令実行時 PRINT命令実行時 ブレーク状態	次のラインを実行(1ラインずつ実行して停止します。)	押している間、実行しているラインあるいは実行したラインを表示
	プログラム実行時のエラー状態		押している間、エラー発生のラインを表示
	トレースモードオン状態	デバッグ動作を実行	押している間、実行しているラインあるいは実行したラインを表示
PRO（スイッチをPROモードに切換え、プログラムラインが表示されていないとき）			
	プログラムの一時停止中	実行を停止しているラインを表示	同　左
	エラー発生後	エラー発生のプログラムラインを表示	同　左
	その他	先頭ラインを表示	末尾ラインを表示
PRO（プログラムラインが表示されているとき）			
		次のプログラムラインを表示	1ライン前のラインを表示

- ENTER キーは表示上ではスペースと同じです。
- キー入力待ちの状態で、約11分間キー入力がなければ自動的に電源が切れます。
（Auto power off：オートパワーオフ機能）

◉ **表示シンボル**

表示部の上側に表示されるシンボルの意味を示します。

BUSY ：プログラムを実行していることを示します。

P ：プリントモードが指定されていることを示します。

DEF ：[DEF] キーが押されたことを示します。このシンボルが表示されているとき、アルファベットキーなどが定義付けキーとして働きます。

DEG ：ディグリー〔°〕
RAD ：ラディアン〔rad〕
GRAD ：グラード 〔g〕

三角関数や逆三角関数の計算において、取り扱う角度の単位を示すシンボルです。

（90° $=\frac{\pi}{2}$〔rad〕=100g ）

SHIFT ：[SHIFT] キーが押されたことを示します。このシンボルが表示されているときは、キーの上側のパネルに茶色で記されている機能が指定されています。

PC-1250/PC-1251用プログラムを使用する場合の注意点

PC－1245とPC－1250／PC－1251のBASIC命令は同一になっていますので、相互に実行させることができますが、PC－1245はPC－1250／PC－1251より表示桁数が少ないため、PC－1250／PC－1251用プログラムをPC－1245で実行させた場合、PRINT、PAUSE、INPUT命令など、表示に関連した部分で、エラーが発生したり、一部表示できない部分が出るなど、不都合な場合があります。
したがって、この場合はプログラムを変更する必要があります。(PC－1251用プログラムが長い場合はPC－1245に入り切らない場合があります。)

(例)

(1) PRINT式 でUSING指定が総桁で16桁を越えているときは ERROR 7 になります。(ただし、整数部の指定桁が11桁を越えている場合は、越えた部分が無視されます。)

17桁以上

```
10 USING"########.########"
20 PRINT A
```

●この場合はUSING指定を16桁以内にしてください。

(2) PRINT式，式 でUSING指定が整数部で(符号、小数点を含む)8桁を越えている場合はERROR 7 になります。

9桁以上

```
10 USING"#########.##"
20 PRINT A,B
```

●この場合はUSING指定の整数部を8桁以内にしてください。

(3) INPUT"……"；A の形でメッセージが15桁を越える場合、越えた桁数だけメッセージの先頭部分が表示されません。

●この場合、メッセージを短かくしてください。

(4) その他、次のような場合は越えた内容が表示されません。

PRINT { 式 / "文字" / 文字変数 } ; { 式 / "文字" / 文字変数 } ; ……

この形で表示内容が16桁を越える。

PRINT { "文字" / 文字変数 } , { "文字" / 文字変数 }

この形で、左あるいは右の文字列が8桁を越える

INPUT"……", A

この形でメッセージが16桁を越える。

●この場合は、表示内容が越えないように編集し直してください。

電源について

本機は本体のみで使用するときは、リチウム電池のみで動作しますが、CE-125またはCE-126Pが接続されている場合、CE-125側(CE-126P側)の電圧が十分あり、リチウム電池の電圧がさがっているときはCE-125側(CE-126P側)の電源からも電気が供給されます。

電池の交換時期

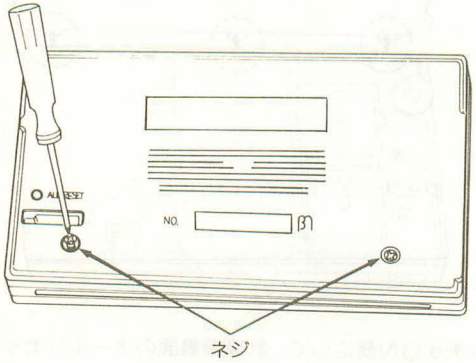
計算機右側面にあるコントラストボリュームを濃くなるほうに一ぱいに回しても、正面から見た表示が薄く、見えにくくなった場合は電池の消耗を示していますので、速やかに新しい電池と交換してください。

(別売の周辺機器CE-125をお持ちの場合またはCE-124、CE-126Pとテープレコーダーをお持ちの場合は、事前にプログラムやデータをテープに記録してください。)

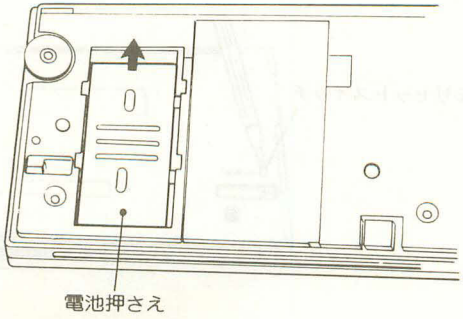
電池を入れる場合、および交換する場合は次の手順にしたがっておこなってください。

(1) 電源／モードスイッチをＯＦＦ位置にしてください。

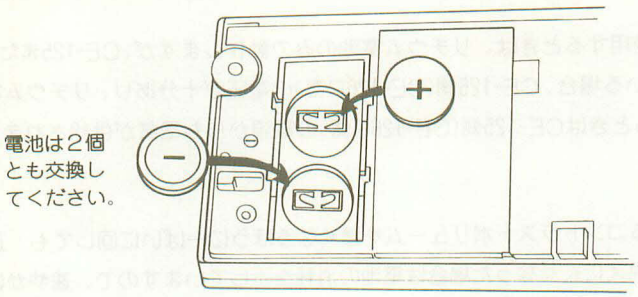
(2) 小型のネジ回しで裏ぶたの2本のネジをはずしてください。



(3) 電池押さえを図の矢印の方向に動かして取りはずしてください。

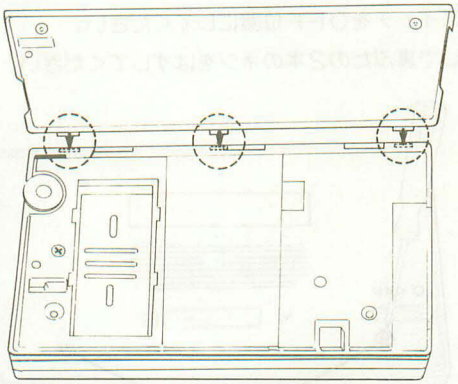


(4) 使用ずみの電池を取りはずし、新しいリチウム電池を2個入れます。
乾いた布で電池をよくふいて、プラス・マイナスをまちがえないように入れてください。

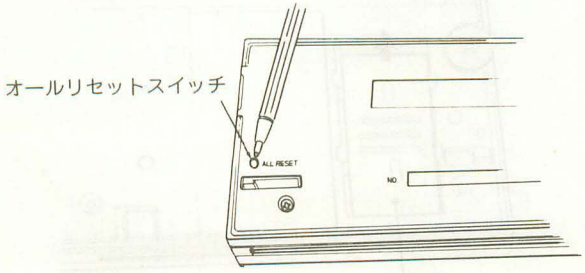


(5) 電池押さえを(3)と逆に取りつけてください。

(6) 裏ぶたを取りつけます。
裏ぶたのツメを本体のミゾに掛けてから裏ぶたを本体に合わせ、軽く押さえながらネジで留めてください。



(7) 電源／モードスイッチをON側にして、計算機裏面のオールリセットスイッチを押し、計算機内の状態をすべてリセットしてください。



そして、そのあとプロンプト記号が表示されることを確認してください。

プロンプト記号が表示されることを確認してください。

電池交換は前記の手順でおこなっていただくか、あるいはお買上げの販売店もしくは別紙一覧表に記載のサービス会社にお申しつけください。有料にて交換いたします。

◆電池は次の種類の新しい電池を２個ご使用ください。

リチウム電池　CR2032

〈電池使用上のお願い〉

1．電池は必ず同時に２個とも交換してください。
2．新しい電池と一度使用した電池をまぜて使用しないでください。
3．電池は必ず同種のものをご使用ください。
4．電池の＋、－の向きを本機の表示通り正しく入れてください。

〈ご注意〉

- **電池は幼児の手がとどかないところに置いてください。**
  万一飲み込んだ場合にはただちに医師にご相談ください。
- 計算機に内蔵された最初の電池は工場出荷時に組み込んでいますので約300時間に満たないうちに電池がなくなることがあります。
- 消耗した電池をそのままにしておきますと、液もれにより計算機をいためることがあります。
- 電池を火中に投入しないでください。火中に投入すると破裂するおそれがあります。

仕　様

形　　名	PC-1245	
計算桁数	10桁（仮数部）＋2桁（指数部）	
計算方式	数式通り（優先順位判別機能つき）	
プログラム言語	BASIC（ベーシック）	
CPU	CMOS　8ビットCPU	
システムROM	24キロバイト（24KB）	
メモリー容量	システムエリア	約0.5キロバイト
	データ専用エリア	208バイト
	プログラム・データエリア	1486バイト
スタック	サブルーチン用	10段
	FOR−NEXT用	5段
	ファンクション用	16段
	データ用	8段
基本計算機能	基本計算　：　加減乗除算	
	関数計算　：　三角関数、逆三角関数、対数、指数、角度変換、べき乗、開平計算、整数化、絶対値、符号関数、円周率	
編集機能	カーソル左右シフト（▶，◀）	
	挿　入（INS）	
	削　除（DEL）	
	ラインアップ・ダウン（↓，↑）	
メモリー保護	バッテリーバックアップ（プログラム、データなど電源オフ時内容保護）	
表　　示	5×7ドットマトリックス　16桁液晶表示	
使用温度	0℃〜40℃	
電　　源	6V⎓（DC）：リチウム電池〈CR2032〉2個	
電池使用時間	約300時間（使用温度20℃で55555555を表示させている場合。電池の使用方法などにより多少の変動があります。）	
消費電力	0.03W	
外形寸法	幅135mm×奥行70mm×厚さ9.5mm	
重　　量	115g（電池含む）	
付属品	ハードカバー、テンプレート1枚、リチウム電池2個(計算機に内蔵)、取扱説明書、入門書、ネームラベル、シャープサービス・お客様ご相談窓口一覧表	

アフターサービスについて

◆修理を依頼されるときは

修理は、お買いあげの販売店に、本機の品名(計算機)、形名(PC-1245)および故障状態をくわしくご説明いただいて、お申しつけください。

ご転居・ご贈答品などでお買いあげの販売店に修理を依頼することができない場合は、もよりのシャープお客様ご相談窓口にお問い合わせください。

◆製品の保証について

① この計算機は取扱説明書の巻末に保証書がついています。保証書は販売店にて所定事項を記入いたしますので、内容をよくお読みのうえ大切に保存してください。

② 保証期間は保証書に記載されている期間です。
保証書の記載内容によりお買いあげの販売店または当社サービス会社が修理いたします。
くわしくは保証書をご覧ください。

③ 保証期間中の修理などアフターサービスについておわかりにならない場合は、お買いあげの販売店、またはもよりのシャープお客様ご相談窓口にお問い合わせください。

④ 保証期間経過後の修理については販売店にご相談ください。修理によって機能が維持できる場合はお客様のご要望により有料修理いたします。

◆お問い合わせは

この製品についてのご意見、ご質問はもよりのお客様ご相談窓口(技術センター)へお申しつけください。付属の「シャープサービス・お客様ご相談窓口一覧表」のとおり、全国にお客様ご相談窓口(技術センター)を設けております。

▶ポケットコンピュータ友の会について◀

入会手続

下記に御紹介するプログラム集の巻頭にある、「ポケットコンピュータ友の会のご案内」での要領に従がい皆様の作成されました興味あるオリジナルプログラム一編以上をお送り下さい。

会員特典

- 会員証を発行致します。(有効期間１年)
- 送付していただいたプログラムは、選考の上、製本化し、出版社を通じて出版・販売いたします。なお、採用プログラム分につきましては記念品を贈呈させて頂きます。
- 上記の本を作る際の選考該当期間中にオリジナルプログラムをお送りいただいた会員の皆様にはこの本を１冊贈呈いたします。

送付先　〒639-11　奈良県大和郡山市美濃庄町492
シャープ株式会社　産業機器事業本部
OA事業推進本部　ポケットコンピュータグループ

既刊プログラム集のご案内

- BASIC・ソフトウェア集No.1(PC-1245用)
- BASIC・ソフトウェア集No.2(PC-1251/1245用)
- BASIC・ソフトウェア集No.4(PC-1245/その他)

事務分野や数学，統計，技術さらにはゲーム分野などで実用性と楽しさ、また、はん用性に優れたプログラムを編集し、出版社を通じて出版・販売いたしております。
詳しくは、お近くの販売店または有名書店にお問い合せください。

CE-125付属のマイクロカセットテープ御利用に際して

PC-1245をプリンター／マイクロカセットレコーダCE-125と接続して御利用の場合に、CE-125に付属のマイクロカセットテープは、そのままでは御利用頂けません。このテープにはPC-1250／1251用にプログラムが21本収録されており、そのうち10本はPC-1250/1251取扱説明書に、他の11本はCE-125の取扱説明書に、プログラム内容及び使用方法が説明されております。これらのプログラムを御利用の場合の注意点、変更点について、以降の項に説明致します。

① PC-1250 / 1251の取扱説明書の事例集に記載されているプログラムは、PC-1245用に変更した説明とプログラムリストを本取扱説明書に載せています。上記マイクロカセットテープより読み込んだ後、修正又は追加の必要なプログラムリストの行には、アンダーラインを引いていますので参考にしてください。

② CE-125の取扱説明書の事例集に記載されているプログラムは、修正点のみを載せています。

③ マイクロカセットテープに収録されているプログラムタイトル及びテープカウンタ等は、CE-125取扱説明書の21ページを参照して下さい。

④ 入門書“ポケットBASIC入門”には、ここに説明するプログラムと同じ内容を持ったプログラムが数本あり、それらについては説明を簡略化していることがあります。
詳細については、個々の説明を参照して下さい。

⑤ CE-125取扱説明書では、左カッコ「（」(または右カッコ「）」）の入力のために [SHIFT] [↓] （または [SHIFT] [↑] ）と説明している箇所があります。これらは [SHIFT] [1] （または [SHIFT] [2] ）と操作を変えて下さい。

（なお、当社はこれらのプログラムの使用によって生ずる金銭上の損害及び逸失利益については、一切の責任を負いませんので、あらかじめご了承ください。）

（目　次）

（ページ）

- ニュートン法による方程式の根…… 110
- 平均・分散・標準編差…… 112
- 円と直線の交点…… 113
- 預金金利計算…… 115
- 日数計算…… 118
- タイプ練習…… 120
- 軟着陸ゲーム…… 120
- 記憶力チェック…… 122
- 追いかけマン…… 124
- ダブルローテーション…… 127

（下記のプログラムはCE-125取扱説明書に掲載されており、変更点のみ記しています。
下記以外はプログラムの変更をせずに利用できます。）

- 行列の積…… 129
- シンプソン法による数値積分…… 129
- 相関係数，1次回帰…… 129
- 度数分布図…… 130
- 並びかえ…… 130
- バイオリズム…… 130
- 英単語暗記…… 131
- プリンタ人物画…… 131

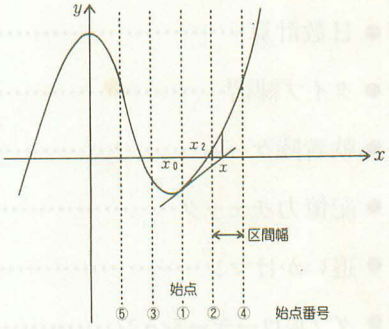
（注）PC-1245に付属の「ポケットBASIC入門」に記載されているプログラムの「ローン返済回数・金額計算」と「分類(SORT)」は、それぞれ、CE-125の取扱説明書に説明されている「借入限度額・ローン返済回数の計算」と「並びかえ」のプログラムに似ていますが、操作およびリスト等で異なる所がありますので、それぞれの説明を参照してください。

PROGRAM TITLE

ニュートン法による方程式の根

簡単明解、根の算出！

一般に方程式の根を求めるのは、やっかいなものですが、ここではニュートン法を用いて根を近似して行く方法をとり上げています。１ツの根が求まると指定した区間幅に応じてニュートン法を用いる始点が自動的に変わって行きます。このプログラムを使って面倒な方程式の根もかんたんに求めることができます。



■使い方

1．DEF A でプログラム開始です。
2．始点、微小区間、区間幅をそれぞれ入力します。
3．根の値を出力します。さらに ENTER を繰り返し押すことにより、始点を先に入力した区間幅ごとに変えていって根を求めます。

■参考（計算内容など）

関数 $y=f(x)$ の $x=Xn$ における接線が x 軸と交わる時の x の値を $Xn+1$ とする

$$Xn+1=Xn-\frac{f(Xn)}{f'(Xn)}$$

となります。

Xn と $Xn+1$ の差の絶対値が 10^{-8} 以下になれば Xn を根とみなし表示します。ここでは１次微分の f'(X) を次の様に定義します。

$$f'(X)=\frac{f(X+h)-f(X)}{h}$$ （hは微小区間）

10^{-8} を変更する時は、340ラインの 1E－8 を変更してください。

■例　題

$x^3-2x^2-x+2=0$（根は－１，１，２）

始点＝０　　微小区間＝10^{-4}　　区間幅＝0.5

として計算します。

関数は、サブルーチンとして、500ライン以後に書き込んで下さい。

書き込み方（例題の場合）

１．PROモードにする。
２．500B＝((X－２)＊X－１)＊X＋２ ENTER
　　510RETURN ENTER

■キー操作手順

ステップ	キー操作	表　示	備　考
1	DEF　A	SHITEN=_	始点の入力待ち
2	0　ENTER	BISHOU KUKAN=_	微小区間の入力待ち
3	0.0001　ENTER	KUKAN HABA=_	区間幅の入力待ち
4	0.5　ENTER	2.	根を表示
5	ENTER	1.	ENTER を繰り返し、始点を変えて根を求めます。
6	ENTER	−1.	
7	ENTER	1.	
8	ENTER	−1.	
9	ENTER	−1.	
10	ENTER	−1.	
11	ENTER	2. ⟆ 以下、同様に始点を変えて根を求められます。	

■プログラムリスト

```
 10:"A": INPUT "SHITEN="
    ;V
 20:INPUT "BISHOU KUKAN=
    ";A
 30:INPUT "KUKAN HABA=";
    W
 40:G=V:F=V:Z=0
 50:IF Z=0 GOTO 70
 60:G=G-W:C=G: GOTO 80
 70:C=G:Z=1
 80:GOSUB 300
 90:F=F+W:C=F
100:GOSUB 300
110:GOTO 50
120:END
300:X=C: GOSUB 500
310:Y=B:X=A+C
320:GOSUB 500
330:D=C:C=D-A*Y/(B-Y)
340:IF ABS (D-C)>=E-8
    GOTO 300
350:BEEP 3: PRINT C
360:RETURN
500:B=((X-2)*X-1)*X+2
510:RETURN
```

■メモリ内容

変　数	内　容
A	微小区間
B	f(X)
C	X_0
D	f(x＋h)
F	√
G	√
V	始点
W	区間幅
X	X
Y	f(x)
Z	初期フラグ

PROGRAM TITLE

平均・分散・標準偏差

このプログラムは入門書“ポケットBASIC入門”の同名のプログラム例と同じ内容を持っています。表示と操作方法は同じですが、プログラムリストは、行番号や変数の使い方などで異なっている所があります。

CE-125付属のマイクロカセットテープよりこのプログラムを読み込んだ後、アンダーラインの箇所を変更すれば、入門書のプログラムと同様に操作できますので、そちらを見てください。

■プログラムリスト

```
 10:"A": CLEAR : WAIT 0
 20:INPUT "DATA SUU=";P
 30:INPUT "OMOMI ARI=1/N
    O=2",A
 40:IF A=2 DIM X(P-1):
    GOTO 70
 50:IF A=1 DIM X(P-1),F(
    P-1)
 70:FOR I=0 TO P-1
 80:B$="X("+ STR$ (I+1)+
    ")="
 85:PAUSE B$: INPUT X(I)
    : GOTO 100
 90:GOTO 85
100:IF A=2 GOTO 150
120:B$="F("+ STR$ (I+1)+
    ")="
130:PAUSE B$: INPUT F(I)
    : GOTO 150
140:GOTO 130
150:NEXT I: END
200:"B": WAIT :I=0
210:B$="X("+ STR$ (I+1)+
    ")=":J=1: PRINT B$;X
    (I)
230:IF A=1 LET B$="F("+
    STR$ (I+1)+")=":
    PRINT B$;F(I):J=2
240:I=I+1
250:IF I=P END
255:GOTO 210
260:"C": PAUSE B$: IF
    LEFT$ (B$,1)="X"
    INPUT "TEISEI CHI ?=
    ";X(I): GOTO 290
270:IF LEFT$ (B$,1)="F"
    INPUT "TEISEI CHI ?=
    ";F(I): GOTO 290
280:GOTO 250
290:IF J=1 GOTO 230
291:GOTO 210
300:"D":N=0:T=0:S=0: FOR
    I=0 TO P-1:X=X(I)
305:F=1: IF A=1 LET F=F(
    I)
310:N=N+F:T=T+F*X:S=S+F*
    X*X: NEXT I
400:WAIT :X=T/N:Q=(S-N*X
    *X)/(N-1):S=√Q
405:PRINT "SOUWA=":
    PRINT T: PRINT "HEIK
    IN=": PRINT X
410:PRINT "BUNSAN=":
    PRINT Q: PRINT "HYOJ
    UN HENSA=": PRINT S:
    END
```

■メモリ内容

変　数	内　容
A	重みありなしプラグ
B$	√
I	√
J	フラッグ
N	√
P	データ数
Q	分　散
S	標準偏差
T	総　和
X	平　均
G(P−1)	データ xi
F(P−1)	データ fi

PROGRAM TITLE

円と直線の交点

平面に与えられた円と直線。その関係はいかに？

平面上に与えられた直線と円の交点を求めるプログラムです。直線の指定のし方は直線上の2点を指定する場合と、直線の1点1方向角を指定する場合とがあります。

■使い方

1．DEF A 操作⇨2点が既知の場合

円の中心の座標(x_0, y_0)円の半径r、直線上の2点の座標(x_1, y_1)、(x_2, y_2)をそれぞれ入力していくと、交点の座標(x_p, y_p)(x_q, y_q)が表示され処理終了です。

2．DEF B 操作⇨1点1方向角が既知の場合

円の中心の座標(x_0, y_0)円の半径r、直線上の1点の座標(x_1, y_1)直線の方向角αをそれぞれ入力していくと、交点の座標(x_p, y_p)(x_q, y_q)が表示され処理終了です。

3．角度は、度、分、秒で123度14分23秒＝123.1423というふうに入力してください。

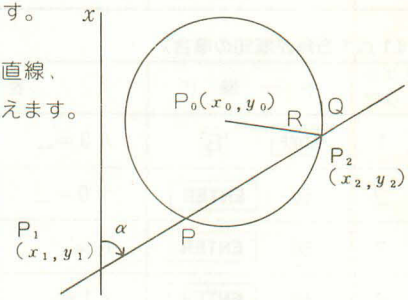
■参考(計算内容など)

○円は、中心の座標(x_0, y_0)と半径rで与えます。

○直線は、2点$P_1(x_1, y_1)P_2(x_2, y_2)$を通る直線、または1点$P_1$を通り方向角$\alpha$の直線として与えます。

○直線と円の2交点をP、Qとします。

○座標値は、小数点以下5桁目を切り捨て処理しています。



■例題

(x_1, y_1)＝(−50, 0)

(x_2, y_2)＝(50, 100)

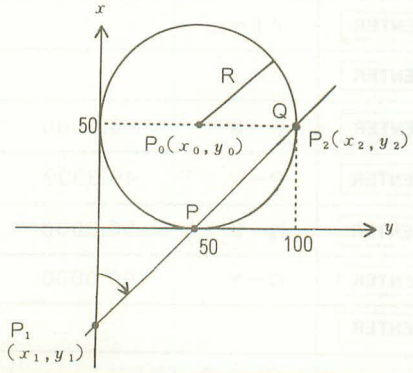
(x_0, y_0)＝(50, 50)

R＝50

α＝45°

(x_p, y_p)＝(0, 50)

(x_q, y_q)＝(50, 100)



■キー操作手順

《2点が既知の場合》

ステップ	キー操作	表示	備考
1	DEF A	X 0＝_	円の中心x座標の入力待ち
2	50 ENTER	Y 0＝_	円の中心y座標の入力待ち
3	50 ENTER	R＝_	円の半径の入力待ち
4	50 ENTER	X1＝_	直線の既知点x座標の入力待ち
5	−50 ENTER	Y1＝_	直線の既知点y座標の入力待ち
6	0 ENTER	X2＝_	直線の既知点x座標の入力待ち
7	50 ENTER	Y2＝_	直線の既知点y座標の入力待ち
8	100 ENTER	P−X 0.0000	交点の座標(x_p, y_p)を表示
9	ENTER	P−Y 49.9999	
10	ENTER	Q−X 50.0000	交点の座標(x_p, y_p)を表示
11	ENTER	Q−Y 100.0000	
12	ENTER	〉	処理終了

《1点1方角が既知の場合》

ステップ	キー操作	表示	備考
1	DEF B	X 0＝_	円の中心x座標の入力待ち
2	50 ENTER	Y 0＝_	円の中心y座標の入力待ち
3	50 ENTER	R＝_	円の半径の入力待ち
4	50 ENTER	X 1＝_	直線の既知点x座標の入力待ち
5	−50 ENTER	Y 1＝_	直線の既知点y座標の入力待ち
6	0 ENTER	A＝_	直線の方向角の入力待ち
7	45 ENTER	P−X 0.0000	交点の座標(x_p, y_p)を表示
8	ENTER	P−Y 49.9999	
9	ENTER	Q−X 50.0000	交点の座標(x_q, y_q)を表示
10	ENTER	Q−Y 100.0000	
11	ENTER	〉	処理終了

■プログラムリスト

```
 10:"A":J=0: GOTO 30
 20:"B":J=1
 30:DEGREE : INPUT "X0=
    ";A,"Y0= ";B,"R= ";C
 40:INPUT "X1= ";D,"Y1=
    ";E
 50:IF J<>0 INPUT "A= ";
    H:H= DEG H: GOTO 90
 60:INPUT "X2= ";F,"Y2=
    ";G
 70:X=F-D:Y=G-E: GOSUB 5
    00
 80:H=X
 90:X=A-D:Y=B-E: GOSUB 5
    00
100:K=W* SIN (X-H)
110:L= ACS (K/C)
120:M=H-90-L:N=H-90+L
130:GOSUB 600
140:USING "#######.####":
    PRINT "P-X";O: PRINT
    "P-Y";P
150:M=N: GOSUB 600
160:PRINT "Q-X";O: PRINT
    "Q-Y";P
170:END
500:W=√(X*X+Y*Y)
510:X= ACS (X/W): IF Y<0
    LET X=360-X
520:RETURN
600:O=A+C* COS M:P=B+C*
    SIN M: RETURN
```

■メモリ内容

変 数	内 容
A	X 0
B	Y 0
C	R
D	X_1
E	Y_1
F	X_2
G	Y_2
H	Q_2°
J	判別フラッグ
K	h
L	d
M	Q p
N	Q_q
O	X_p、X_q
P	Y_p、Y_q
W	L
X	△X、θ
Y	△Y

PROGRAM TITLE

預金金利計算

利子で暮らせる日が来るかも？

貯金をするのもいいですが、一年間を通して利子はいくらぐらいになっているだろう、と考えることがあります。そこでこのプログラムを利用し、元利、利息、総元利等を求めようというものです。知っておいて損はない金利計算。あなたも利子で暮らせる日が来るかもしれないですよ。

■使い方

1．DEF A 操作⇨総合計算出と初期設定

ENTER キーをくり返し押して、合計メモリーがクリアされていることを確認してください。

2．DEF C 操作⇨利率設定

利率を入力します。

3. DEF A 操作⇨利息計算

元金を入力し、続いて日数を入力すると、利息が表示され、さらに元利合計が表示されます。引き続いて、元金入力からの操作を行ない、利息計算を繰り返します。必要な利息計算がすべて終了すると次の操作へ移ってください。

4. DEF B 操作⇨総合計算出と初期設定

総元金計、総利息合計、総元利合計をそれぞれ算出し、合計メモリーをクリアしたあと、計算終了です。

■計算内容など

一般式：利息＝元金×$\frac{日数}{36500}$×利率　　Σ元利合計＝Σ元金＋Σ利息

■例題

次のデータに基き、預金金利計算を行ないます。

利率、元金、日数を入力し、利息と元利の合計を求めます。

付利単位百円、円未満切捨て　利率7(%)

元　金（円）	日数	利　息（円）	元利合計（円）
50,000	60	575	50,575
45,000	125	1,078	46,078
789,000	175	26,480	815,480
合計 884,000		28,133	912,133

■キー操作手順 《初期設定》

ステップ	キー操作	表示	備考
1	DEF B	TOTAL GANKIN	元金の合計を表示
2	ENTER	0.	
3	ENTER	TOTAL RISOKU	利息の合計を表示
4	ENTER	0.	
5	ENTER	TOTAL GANRI. KEI.	元利合計の合計を表示
6	ENTER	0.	
7	ENTER	CLEAR OK	
8	ENTER	〉	処理終了

《利率設定》

ステップ	キー操作	表示	備考
1	DEF C	RIRITSU(%)=_	利率の入力待ち
2	7 ENTER	>	処理終了

《利息計算》

ステップ	キー操作	表示	備考
1	DEF A	GANKIN=_	元金の入力待ち
2	50000 ENTER	NISSU=_	日数の入力待ち
3	60 ENTER	RISOKU	利息の表示
4	ENTER	575.	
5	ENTER	GANRIGOUKEI	元利合計の入力待ち
6	ENTER	50575.	
	ENTER	GANKIN=	
≀		≀	
		以下、同様に繰り返します。	
15	ENTER	GANKIN=_	元金の入力待ち
16	DEF B	TOTAL GANKIN	
17	ENTER	884000.	
18	ENTER	TOTAL RISOKU	総利息合計を表示
19	ENTER	28133.	
20	ENTER	TOTAL GANRI. KEI	総元利合計を表示
21	ENTER	912133.	
22	ENTER	CLEAR OK	メモリー内容のクリア終了
23	ENTER	>	処理終了

■プログラムリスト

```
10:"C": INPUT "RIRITSU(
   %)=";C
20:END
30:"B"
40:PRINT "TOTAL GANKIN"
   : PRINT E
50:PRINT "TOTAL RISOKU"
   : PRINT D
60:PRINT "TOTAL GANRI.K
   EI": PRINT E+D
70:E=0:D=0: PRINT "CLEA
   R OK": END
80:"A": WAIT : USING :
   INPUT "GANKIN=";A
90:INPUT "NISSU=";B
100:X= INT ( INT (A/100)
   *B*C/36500*100)
110:E=E+A
120:D=D+X
130:PRINT "RISOKU":
   PRINT X
140:PRINT "GANRIGOUKEI":
   PRINT A+X
150:GOTO 80
160:END
```

■メモリ内容

数値変数	内　容
A	元　金
B	日　数
C	利　率
D	利息合計
E	元金合計
X	利息

PROGRAM TITLE

日数計算

知りたい日数がすぐ分かる！

あなたが生まれた時から今日まで何日ぐらい経過しているのか、と考えたりすることがあります。このプログラムでは、自分の知りたい、期間の長さを明確に計算してくれます。基準となる年月日と終わりの年月日を打ち込むと、ポケットコンピュータが知りたい日数を算出してくれます。

■使い方

1．[DEF] [A] でプログラムを開始します。基準となる年・月・日をそれぞれ入力します。そして終りの年・月・日をそれぞれ入力すると、日数が表示されます。

2．プログラムを終了させる時には、終了年の入力待ちの時に [DEF] [Z] 操作により処理終了になります。

3．年の入力は、明治(Ｍ)、大正(Ｔ)、昭和(Ｓ)についてはＳ50(昭和50年)のような入力も可能です。他は西暦で入力して下さい。

■例題

昭和50年１月１日から
→1977年３月４日までの日数：793日
→昭和55年６月４日までの日数：1981日

■キー操作手順

ステップ	キー操作	表示	備考
1	DEF A	START YEAR=_	基準年の入力待ち
2	S50 ENTER	MONTH=_	基準月の入力待ち
3	1 ENTER	DAY=_	基準日の入力待ち
4	1 ENTER	END YEAR=_	終了年の入力待ち
5	1977 ENTER	MONTH=_	終了年の入力待ち
6	3 ENTER	DAY=_	終了日の入力待ち
7	4 ENTER	DAYS= 793.	日数を表示
8	ENTER	END YEAR=_	終了年の入力待ち
9	S55 ENTER	MONTH=_	終了月の入力待ち
10	6 ENTER	DAY=_	終了日の入力待ち
11	4 ENTER	DAYS= 1981.	日数を表示
12	ENTER	END YEAR=_	終了年の入力待ち
13	DEF Z	〉	プログラム終了

■プログラムリスト

```
10:"A"
20:INPUT "START YEAR=";
   A$,"MONTH=";S,"DAY="
   ;T: GOSUB 610
25:R=F
30:INPUT "END YEAR=";A$
   ,"MONTH=";V,"DAY=";W
   : GOSUB 610
50:H=R
60:G=S:I=T
70:GOSUB 500
80:J=I
100:H=F
110:G=V:I=W
120:GOSUB 500
130:X=I-J
140:WAIT : USING : PRINT
   "DAYS=",X
150:GOTO 30
500:IF G-3>=0 LET G=G+1:
   GOTO 520
510:G=G+13:H=H-1
520:I= INT (365.25*H)+
   INT (30.6*G)+I
530:I=I- INT (H/100)+
   INT (H/400)-306-122:
   RETURN
610:IF LEFT$ (A$,1)="S"
   LET B=1925: GOTO 670
620:IF LEFT$ (A$,1)="T"
   LET B=1911: GOTO 670
630:IF LEFT$ (A$,1)="M"
   LET B=1867: GOTO 670
640:F= VAL A$
650:RETURN
670:C$= MID$ (A$,2,3):D=
   VAL C$:F=B+D
680:RETURN
690:"Z": END
```

V	終了月
W	終了日
X	求める日数

■メモリ内容

変数	内容
A$	年入力用
B	√
C$	√
D	√
F	年（計算後）
G	√
H	√
I	√
J	√
R	√
S	基準月
T	基準日

PROGRAM TITLE

タイプ練習

（注）このプログラムは入門書“ポケットBASIC入門”の“キータッチゲーム（タイプ練習）”と同じ内容を持っています。CE-125付属のマイクロカセットテープのプログラムリストとの変更点だけを記しますので、プログラム内容及び操作方法は入門書を見てください。

■プログラムリスト

```
70:BEEP 3:E=0: WAIT 30:
   USING "&&&&&&&&&"

80:FOR W=1 TO B*10/L:
   PRINT Y$;A$: IF E=B
   LET W=B*20/L: GOTO 1
   00

160:IF P>X LET X=P: WAIT
   100: PRINT "YOUR SCO
   RE BEST"
```

PROGRAM TITLE

軟着陸ゲーム

燃料が少ない！緊急着陸せよ！

残り少ない燃料でいかにロケットを着陸させるか。ロケットは自由落下する。それに合わせてロケットエンジンを点火せよ。点火するのが早かったり、燃料消費が多過ぎると惑星の塵となってしまい、遅過ぎたりすると惑星面に墜突してしまう。さあ、あなたの腕と度胸でこのロケットを無事に着陸させてみよう。点火時と燃料消費のコントロールが成功させるコツだよ。

■遊び方

1．[DEF] [A] でプログラムをスタートします。

2．[0]～[9]までのキーで消費燃料を入力します。
着陸させる時にはうまく燃料を調整しながら着陸させてください。成功した時は残り燃料、失敗した時は〝GOOD BY!!〟が表示され、再ゲームをするかどうかの入力待ちとなります。

■参考

重力は5m/s² 単位時間(S)に5単位の燃料を消費すると重力を相殺します。

計算式

$H = H_0 + V_0 t + \frac{1}{2} a t^2$

$V = V_0 + at$

$V^2 = V_0^2 + 2aH$

$H_0 = 500$、$V_0 = -50$、$F_0 = 200$

H：高度　　V_0：初速度

V：速度　　H_0：初高度

a：重力加速度　　F_0：初燃料

t：時間　　F：消費燃料

注）プログラムの30ラインの初高度、初燃料とwait時間の数値をかえることによって面白さをかえることができます。

■キー操作手順

ステップ	キー操作	表示	備考
1	DEF　A	＊＊＊START＊＊＊	ゲーム開始
2	9	500 −50 200 0	高度・速度・残り燃料・消費燃料の表示
3	9	452 −46 191 9	
		⟆ 繰り返します。	
		SEIKO！！	成功の場合
		NOKORI　F=10.	残り燃料の表示
		REPLAY(Y/ N)？	再ゲームするかどうか選択
31	Y		ステップ1へもどる
	N	>	プログラム終了

■プログラムリスト

```
10:"A": WAIT 50: CLEAR
   : USING :S=-50:A=0:D
   $=""
20:BEEP 3: PRINT "  ***
    START ***"
30:DATA "JIKAN=",50,"NE
   NRYOU=",400,"KODO=",
   500
40:RESTORE
50:READ B$,W,B$,F,B$,H
60:WAIT W
70:PRINT USING "####";H
   ;S;F;C
80:IF F<=0 GOTO 170
90:BEEP 1:D$= INKEY$
100:IF D$="" LET C=A:
   GOTO 130
110:C= VAL D$
120:A=C
130:IF C>F LET C=F
140:F=F-C:X=C-5:H=H+S+X/
   2:S=S+X
150:IF H>0 GOTO 70
160:IF ( ABS H<5)+( ABS
   S<5)=2 BEEP 5: PRINT
   "SEIKO!!": GOTO 180
170:BEEP 3: PRINT "GOOD
   BY !!": GOTO 190
180:WAIT 150: PRINT
   USING "####";"NOKORI
    F=";F
190:INPUT "REPLAY (Y/N)
   ?";Z$
200:IF (Z$="Y")+(Z$="N")
   <>1 GOTO 190
210:IF Z$="Y" GOTO 10
220:END
```

■メモリ内容

変　数	内　容
A	√
B$	√
C	消費燃料
D$	消費燃料
F	初燃料、残り燃料
H	初高度、高度
S	速　度
W	WAIT時間
X	√
Z$	√

PROGRAM TITLE

記憶力チェック

あなたは天才になれるか？

このプログラムは、12桁の文字列を記憶し、それを答えるものです。表示時間は約5秒。あなたの記憶力はどのくらいの評価を得られるかな？ただの凡人で終るか天才になれるか、それはあなたの記憶力しだいでスタートします。

■遊び方

1．DEF A でタイトルが表示されます。

2．下図のように、英字2文字、数字2文字、計4文字の列が3ヶ表示されます（約5秒間）

英英数数	英英数数	英英数数
←イ→	←ロ→	←ハ→

3．イ、ロ、ハ、の文字列を記憶し、イ、ロ、ハ別に記憶している内容を答えてください。

4．回答内容の分析により、記憶力を7ランクに分け、評価メッセージが表示されます。
点数は、イ、ロ、ハ個々に上、下3桁ずつ分け全問正解の時は6点になります。

点数	評価メッセージ
0	MANUKE
1	DAME
2	BONJIN
3	RYOUKOU
4	YUUSHUU
5	＊ SHUUSAI＊
6	＊＊TENSAI＊＊

■キー操作手順

ステップ	キー操作	表示	備考
1	DEF A	KIOKURYOKU CHECK	タイトル表示
2		㋑ AB12 ㋺ PK43 ㋩ VC72	問題文字列を表示(5秒間)
3		ANS. ＝_	㋑の文字列入力待ち
4	AB12 ENTER	ANS. ＝_	㋺の文字列入力待ち
5	PK73 ENTER	ANS. ＝_	㋩の文字列入力待ち
6	VA77 ENTER	AB12 PK73 VA77	記憶した文字列表示
		AB12 PK43 VC72	問題文字列の表示(1.5秒)
		AB12 PK73 VA77	記憶した文字列表示
		RYOUKOU	記憶力の評価メッセージ
		＊REPLAY(Y／N)?_	リプレイの入力要求
7	Y 又は N ENTER		Yならステップ2へ 以外ならEND

■プログラムリスト

```
10:"A": USING : WAIT 200: PRINT "KIOKURYOKU CHECK": CLEAR : RANDOM
20:DIM G$(6)*1,N$(10)*1,V$(3)*2,X$(3)*4,Z$(3)*2,Y$(3)*4
30:FOR I=1 TO 9:N$(I)=STR$ I: NEXT I:N$(10)="0"
50:FOR I=1 TO 6
60:J= RND 26:J=J+64
70:G$(I)= CHR$ (J): NEXT I
80:FOR I=1 TO 3
90:Y$(I)=" "
100:FOR J=1 TO 2:K= RND 9
110:Y$(I)=Y$(I)+N$(K): NEXT J
120:J=(I-1)*2+1
130:A$(I)=G$(J)+G$(J+1)
140:H$=Y$(I):A$(I+3)= RIGHT$ (H$,2): NEXT I
150:GOSUB 500
160:FOR I=1 TO 3
170:INPUT "  ANS. = ";X$(I):X$(I)= LEFT$ (X$(I),4)
180:Z$(I)= LEFT$ (X$(I),2)
190:V$(I)= RIGHT$ (X$(I),2): NEXT I
200:GOSUB 520: GOSUB 500: GOSUB 520
210:N=0
220:FOR I=1 TO 3
230:IF A$(I)=Z$(I) LET N=N+1
240:IF A$(I+3)=V$(I) LET N=N+1
250:NEXT I
260:N=N+1
270:WAIT 150: ON N GOTO 300,310,320,330,340,350,360
300:BEEP 1: PRINT "  MANUKE": GOTO 370
310:BEEP 1: PRINT "  DAME": GOTO 370
320:BEEP 2: PRINT "  BONJIN": GOTO 370
330:BEEP 2: PRINT "RYOUKOU": GOTO 370
340:BEEP 3: PRINT " YUUSHUU!": GOTO 370
350:BEEP 4: PRINT "* SHUUSAI *": GOTO 370
360:BEEP 5: PRINT "**TENSAI**"
370:W$="": BEEP 1: INPUT "REPLAY (Y/N)?";W$
380:IF W$="N" THEN 600
390:IF W$="Y" THEN 50
395:GOTO 370
400:GOTO 370
500:WAIT 300: BEEP 2: PRINT A$(1);A$(4);" ";A$(2);A$(5);"  ";A$(3);A$(6)
510:RETURN
520:WAIT 80: BEEP 1: PRINT USING "&&&&&&";X$(1);X$(2);X$(3)
525:USING
530:RETURN
600:END
```

■メモリ内容

変　数	内　容
A$	2桁文字
B$	
C$	
D$	
E$	
F$	
H$	√
I	添　字
J	乱数発生
K	乱数発生
L	乱数発生
N	カウンター
W$	REPLAY　入力用
G$(6)	英　字
N$(10)	WORK
V$(3)	回答後2桁
X$(3)	WORK
Y$(3)	WORK
Z$(3)	回答用3桁

PROGRAM TITLE

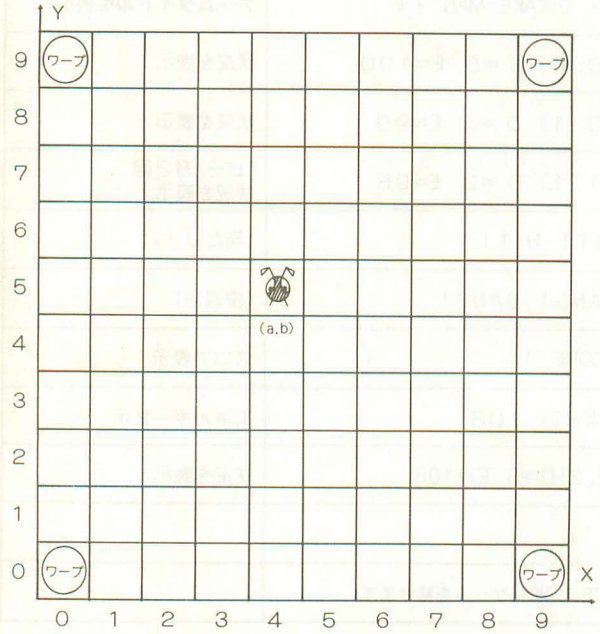
追いかけマン

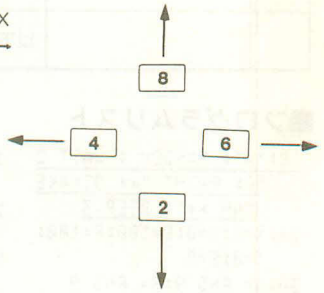
悪戦苦闘！追いかけマン。

今日も追いかけマンが、虫を退治しにやって来た。虫は、ありとあらゆる手を使い逃げ回る。追いかけマンは、このすばしっこい虫どもを退治することができるだろうか。エネルギーには限界がある。エネルギーのなくならないうちににげまどう虫をやっつけろ。

追いかけマンの位置(X，Y)
虫の位置　　(a，b)

虫の位置が角に来た時は、ワープ
することができる。





■使い方

1．DEF A　でゲーム開始です。

2．追いかけマンの移動はキー 2 4 6 8
を使って、右図のように動きます。

3．追いかけマンが位置をかえれば虫も上下左右に１つ動きます(そのままの位置でも可)
追いかけマンと虫の位置は、

(x , y)	D＝ℓ	E＝e
⇩ 現在位置座標	⇩ ヒント (距離差)	⇩ 残りエネルギー

と表示され、ヒント(距離差)がＡＢＳ(Ｘ－a)＋ＡＢＳ(Ｘ－b)の値として表示されます。エネルギーは最初100与えられており、時間がたつにつれて１つずつ減ってゆきます。

4．虫を見つけるとBEEP音が５回鳴ります。退治できればエネルギーが(５、10、15)のいずれかが増え、エネルギーが０になるまでにどれだけ虫を退治したかによってゲームを競います。最後にスコアが表示され、ゲーム終了です。

■キー操作手順

ステップ	キー操作	表示	備考
1	DEF A	＊＊ OIKAKE MAN ＊＊	ゲームタイトルを表示
		(0, 0) D=5 E=100	状況を表示
2	8	(0, 1) D=4 E=99	状況を表示
3	6	(1, 1) D=2 E=98	「ピー」音2回 状況を表示
4	8	HIT! HIT!	(当たり!)
		BANG! BANG!	(虫退治)
		SCORE 1	スコア表示
		ENERGY 108	エネルギー表示
		(1, 2) D=6 E=108	状況を表示。
∫		∫	
		以下、同様にゲームを続けます。	

■プログラムリスト

```
10:"A": RANDOM : WAIT 2
   50: PRINT "** OIKAKE
   MAN **": BEEP 3
20:X=0:Y=0:E=100:F=100:
   T=0:S=0
30:A= RND 9:B= RND 9
40:L= ABS (X-A)+ ABS (Y
   -B)
50:IF X=A AND Y=B GOTO
   400
100:IF L=1 BEEP 3
110:IF L=2 BEEP 2
120:IF L=3 BEEP 1
130:WAIT 50: PRINT "(";
   STR$ (X);",";STR$ (
   Y);") D=";STR$ (L);
   " E=";STR$ (E)
150:S=S+1:E=F- INT (S/2)
153:IF E<=0 THEN 500
155:G$= INKEY$ : IF G$="
   " GOTO 130
157:BEEP 1
160:IF G$="2" LET Y=Y-1:
   GOTO 210
170:IF G$="4" LET X=X-1:
   GOTO 210
180:IF G$="6" LET X=X+1:
   GOTO 210
190:IF G$="8" LET Y=Y+1:
   GOTO 210
200:GOTO 150
210:IF X<0 LET X=0: GOTO
   150
220:IF Y<0 LET Y=0: GOTO
   150
230:IF X>9 LET X=9: GOTO
   150
240:IF Y>9 LET Y=9: GOTO
   150
250:IF X=A AND Y=B GOTO
   400
260:E=F- INT (S/2)
270:IF E<=0 GOTO 500
280:R= RND 5
290:IF R=1 LET B=B-1:
   GOTO 340
300:IF R=2 LET A=A-1:
   GOTO 340
```

■メモリ内容

変数	内容
A	虫の位置 X座標
B	虫の位置 Y座標
C	増加エネルギー値
E	残りのエネルギー
F	エネルギー総量
G$	2,4,6,8キー入力
L	虫とおいかけマンとの距離
R	虫の移動値(1～4)
S	経過TIME
T	得点
X	追いかけマンの位置 X座標
Y	追いかけマンの位置 Y座標

```
310:IF R=3 LET A=A+1:
    GOTO 340
320:IF R=4 LET B=B+1:
    GOTO 340
340:IF A<0 OR A>9 GOTO 3
    70
350:IF B<0 OR B>9 GOTO 3
    70
360:GOTO 40
370:BEEP 4: PAUSE "*** W
    ARP ***": GOTO 30
400:PAUSE "HIT! HIT!"
410:BEEP 5
420:PAUSE "BANG!  BANG!"
430:T=T+1:C= RND 3*5:F=F
    +C
435:E=F- INT (S/2)
440:WAIT 100: PRINT "SCO
    RE ";T: PRINT "ENERG
    Y ";E
450:GOTO 30
500:WAIT 100: PRINT "SCO
    RE "; STR$ (T): WAIT
    : PRINT " *GAME OVER
    !!*"
510:END
```

PROGRAM TITLE

ダブルローテーション

素早さが勝負の決め手！

あちらへ動き、こちらへ動き、大忙がしのアルファベット群。ランダムに並んでいる文字配列をアルファベット順に並べ換えよう。素早く並べ換えることが勝負の決め手となる。どこから文字を動かしていくのか、というのも重要なキーポイントとなるよ。

■遊び方

1．DEF A でプログラムを開始すると〝DOUBLE ROTATION〟の表示があり、次にアルファベット(A～J)がランダムに表示されます。

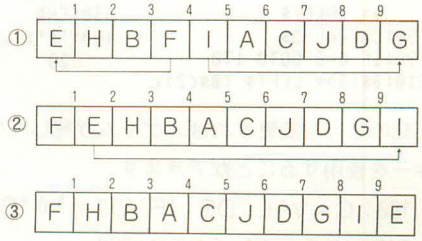
2．文字と文字の間をブレイクポイント(1～9)とし、動かそうとする番号を入力すると、その両側の文字がローテーションされてそれぞれ両端に配置されます。

3．次々と文字を動かしてゆき、並べ終えるとそれまでに要した回数がスコアとして表示されます。
従って、スコアが低いほどよいわけです。

4．同じ初期ランダムアルファベットにより再ゲームをしたい時は DEF B でアルファベット(A～J)が表示されます。後は、DEF A と同様に操作します。

■参考

1のような配列のとき、4を入力するとF、Iはそれぞれローテーションされ②の配列を作ります。次に1を入力するとFはもともと端に位置しているのでEだけがローテーションされて③の配列を作ります。



■キー操作手順

ステップ	キー操作	表示	備考
1	DEF A	DOUBLE ROTATION	
		A～J	
2	1 ～ 9	繰り返します。	1 ～ 9 までを選択しキーを入力
46		ABCDEFGHIJ	
47		GAME END	
48		YOUR SCORE 45	
		〉	
1	DEF B	A～J	DEF A の最初のランダムなアルファベット表示
2	1 ～ 9	繰り返します。	

■プログラムリスト

```
 10:"A": CLEAR : WAIT 50
    : RANDOM : DIM B$(4)
 20:PAUSE "DOUBLE ROTATI
    ON"
 30:B$(0)="ABCDEFGHIJ"
 40:B$(1)=""
 50:A=0
 60:FOR I=1 TO 10
 70:R= RND 10
 80:S=2^(R-1)
 85:B=S AND A
 90:IF B<>0 GOTO 70
100:A=A OR S
110:B$(1)=B$(1)+ MID$ (B
    $(0),R,1): NEXT I
120:B$(2)=B$(1)
130:N=0
150:BEEP 1
170:D$="": PRINT B$(2):D
    $= INKEY$
180:C= VAL D$
190:IF C=0 GOTO 170
210:B$(3)= LEFT$ (B$(2),
    C)
220:B$(4)= RIGHT$ (B$(2)
    ,10-C)
240:IF C=1 GOTO 260
250:B$(3)= RIGHT$ (B$(3)
    ,1)+ LEFT$ (B$(3),C-
    1)
260:IF C=9 GOTO 280
270:B$(4)= RIGHT$ (B$(4)
    ,9-C)+ LEFT$ (B$(4),
    1)
280:B$(2)=B$(3)+B$(4)
290:N=N+1
300:IF B$(2)<>B$(0) GOTO
    150
310:BEEP 5: PAUSE "GAME
    END"
320:WAIT 200: PRINT
    USING "####";"YOUR S
    CORE";N
330:END
400:"B": WAIT 50: GOTO 1
    20
```

■メモリ内容

変数	内容
A	√
B	√
C	√
D$	入力キー
I	√
N	スコア
R	乱数
S	√
B$(4)	アルファベット

下記の2行を変更すれば、ゲームが易しくなり、また、左端のブレイクポイントとして 0 キーを使用することができます。

180：C＝VAL D$：IF D$＝"0"GOTO 210

240：IFC＝＜1 GOTO 260

以降の項には、ＣＥ-125取扱説明書の事例集に載っているプログラムをＰＣ-1245用として、利用する場合の変更点だけを載せています。

PROGRAM TITLE	行列の積	CE-125取扱説明書 23～28ページ

26ページのキー操作手順　ステップ４

4	ENTER	ALL PRINT (Y/N)_	

プログラムリスト 250ライン

```
250:INPUT "ALL PRINT(Y/N
    )";G$
```

PROGRAM TITLE	シンプソン法による数値積分	CE-125取扱説明書 29～35ページ

32ページ，33ページのキー操作手順　ステップ１

1	DEF A	INP.=1/CAL.=2?	データを入力か、組み込んだ計算式をもとに計算するかの選択

32ページのキー操作手順　ステップ50

50	ENTER	ALL PRINT (Y/N)_	データ訂正の終了 すべてのデータをプリントするかの選択

35ページプログラムリスト　10，140ライン

```
 10:"A": CLEAR : WAIT 0:
    INPUT "INP.=1/CAL.=2
    ?";O: GOTO 16

140:INPUT "ALL PRINT(Y/N
    )";W$
```

PROGRAM TITLE	相関係数，１次回帰	ＣＥ-125取扱説明書 36～41ページ

39ページのキー操作手順《入力データの訂正》ステップ４

4	ENTER	ALL PRINT (Y/N)_	データリスト作成の選択

41ページの560ライン

```
560:0$="": INPUT "ALL PR
    INT(Y/N)";0$
```

PROGRAM TITLE	度数分布図	CE-125取扱説明書 42～46ページ

44ページのキー操作手順　ステップ1

1	RUN ENTER	A=<DATA=<B	データの範囲 A≦DATA≦B

46ページのプログラムリスト20ライン

```
20:PRINT "A =< DATA =<
   B": INPUT "A=";A,"B=
   ";B
```

PROGRAM TITLE	並びかえ	CE-125取扱説明書 57～61

59ページのキー操作手順　ステップ3（2箇所）

3	7 ENTER	UP=1/down=2?	UP(小さい順), down (大きい順)の選択

3	5 ENTER	UP=1/down=2?	UP(小さい順), down (大きい順)の選択

```
30:INPUT "UP=1/DOWN=2?"
   ;0
```

PROGRAM TITLE	バイオリズム(セミグラフィック)	CE-125取扱説明書 67～71ページ

69ページのキー操作手順ステップ2

2	SHARP ENTER	SEINEN GAPPI?	生年月日を次に入力します。
		NEN=	誕生年の入力待ち

71ページプログラムリスト30ライン

```
30:WAIT 120: PRINT "SEI
   NEN GAPPI ?": WAIT :
   INPUT "NEN=";U$,"TSU
   KI=";V,"HI=";W
```

PROGRAM TITLE	英単語暗記	CE-125取扱説明書 72〜77ページ

74ページのキー操作手順《データの訂正》ステップ4

4	ENTER	ALL PRINT (Y/N)_	データリストの印字選択

75ページのキー操作手順《データの記録と読み込み》ステップ1の次

		IN=1/OUT=2 ?_	IN(データの読込み)OUT(データの書き込み)の選択

77ページプログラムリスト 130,330ライン

```
130:INPUT "ALL PRINT(Y/N
    )";A$: IF A$="N" END

330:"Z": PAUSE "TAPE SET
     UP !": INPUT "IN=1/
    OUT=2?";A$
```

PROGRAM TITLE	プリンタ人物画	CE-125取扱説明書 78〜80ページ

79ページのキー操作手順ステップ1

1	DEF	JINBUTSU-GA	タイトルを表示

80ページのプログラムリスト 25ライン

```
25:BEEP 3: PAUSE "JINBU
   TSU-GA"
```

エラーコードとエラー内容

エラーコード	エ ラ ー 内 容
1	文法エラー ●文法的に実行できない場合のエラー 例　3＊／2 [ENTER] ●プログラムのみで実行可能な命令をマニュアル操作で、マニュアル操作でのみ実行可能な命令をプログラムで実行しようとした 例　ＰＲＩＮＴ　Ａ [ENTER]（マニュアル操作）
2	演算エラー ●計算結果が１×10^{100}以上になった ●除数が０の除算を実行 ●不合理な演算の実行（ＬＮ-30、ＡＳＮ1.5など） ●その他
3	ＤＩＭエラー、範囲制限エラー ●すでに宣言されている配列変数名を再度宣言している。 ＤＩＭ命令で宣言していない配列変数を使っている。 ＤＩＭ命令で宣言した大きさを越えて配列変数を使っている。 など ●指定の数値が規定されている範囲から外れている。 例　ＤＩＭ　Ｂ（256） [ENTER]（配列変数の配列要素が255を越えている。） 10　ＦＯＲ　Ａ＝１　ＴＯ　32768（ＦＯＲ－ＮＥＸＴ文において、終値、ＳＴＥＰ値が、－32768～32767の範囲を越えている。）
4	ラインエラー ●ＧＯＴＯ、ＧＯＳＵＢ、ＲＵＮ、ＬＩＳＴ、ＴＨＥＮなどの命令で指定された、ラインやラベルが存在しない。 ●プログラム入力時、ラインナンバーが999を越えている。
5	深みエラー ●ＧＯＳＵＢ(10段)、ＦＯＲ－ＮＥＸＴ(5段)での段数オーバー ●ＲＥＴＵＲＮ文に対するＧＯＳＵＢ文がない ＮＥＸＴ文に対するＦＯＲ文がない ＲＥＡＤ文に対するＤＡＴＡ文がない ●データバッファ(8段)をオーバー ファンクションバッファ(16段)をオーバー 文字記憶エリア(80文字)をオーバー
6	容量オーバー ●プログラム及び変数が大きすぎて、プログラム・データエリアの容量を越えた

7	フォーマットエラー ●ＰＲＩＮＴ、ＰＡＵＳＥ、ＬＰＲＩＮＴ命令において数値データを表示する際、ＵＳＩＮＧ命令で指定されたフォーマットで表示できない。
8	入出力エラー プリンタ、テープレコーダの入出力におけるエラー
9	その他のエラー ●数値変数に文字を代入しようとした。また逆に文字変数に数値を代入しようとした、など変数名の不適合 例　Ａ$＝123 [ENTER] Ｂ$＝Ａ＊2 [ENTER] ●パスワードの不一致など

入力ミスに関して

プログラムを実行したとき、プログラムの入力ミスによりエラーになることがあります。この場合、次のことに留意してください。

◎ **入力ミスを訂正するような場合**

（例）ＬＰＲＩＮＴをＫＰＲＩＮＴと入力していたとき

```
                    10：K PRINT A$
[►]                 10 K PRINT A$
L [ENTER]           10：L PRINT A$
                       └ スペースがある
```

このような方法での修正では、計算機には１つの命令として判断されません。
このような場合はＫＰＲＩＮＴを削除して、改めてＬＰＲＩＮＴと入力してください。

```
                    10：K PRINT A$
[►]                 10 K PRINT A$
[SHIFT][DEL◄]…[SHIFT][DEL◄]   10 A$
[SHIFT][INS►]…[SHIFT][INS►]   10 □□□□□□A$
LPRINT [ENTER]      10：LPRINT A$
                       └ １つの命令と判断されたときは
                         スペースは入らない。
```

なお、命令が正しく入力されているかどうかはカーソルでチェックすることができます。

（正しい入力の場合）

```
    10：RADIAN
[►] 10 RADIAN
[►] 10 RADIAN_
```

（まちがった入力の場合）

```
    10：RADAN
[►] 10 RADAN
[►] 10 RADAN
[►] 10 RADAN
```

キャラクタ・コード表

上位ビット(上位桁) →

下位ビット(下位桁) ↓

	0000	0001	0010	0011	0100	0101	0110	0111
16進	0	1	2	3	4	5	6	7
0000	0			0	@	P	匠	
0001	1		!	1	A	Q		
0010	2		"	2	B	R		
0011	3		#	3	C	S		
0100	4		$	4	D	T		
0101	5		%	5	E	U		
0110	6		&	6	F	V		
0111	7		'	7	G	W		
1000	8		(	8	H	X		
1001	9		)	9	I	Y		
1010	A		*	:	J	Z		
1011	B		+	;	K	↙		
1100	C		,	<	L	¥		
1101	D		-	=	M	π		
1110	E		.	>	N	^		
1111	F		/	?	O	_		

注）表中のコード0（&00）のキャラクタはヌル（ＮＵＬ）を示します。
また、コード32（&20）のキャラクタはスペースです。

注）本機は斜線部分にあたるコード、キャラクタは持っていません。
ＣＨＲ$命令により、本機が持っていないキャラクタコードを指定しますと、
実行時エラーになります。

関数・命令(省略形)一覧

1．基本関数

関　数	省略形	備　考
SIN	SI.	sin
COS		cos
TAN	TA.	tan
ASN	AS.	$\sin^{-1}$
ACS	AC.	$\cos^{-1}$
ATN	AT.	$\tan^{-1}$
LOG	LO.	$\log_{10} X$
LN		$\log_e X$
EXP	EX.	e^x
^		べき乗
√‾(SQR)	SQ.	開平
DEG		度分秒→10進数度変換
DMS	DM.	10進数度→度分秒変換
INT		整数化（インテジャー）　INT(10/3)　ENTER　→ 3
ABS	AB.	絶対値（アブソリュート）ABS(−5＊2)　ENTER　→10
SGN	SG.	符号関数（シグナム）
π（PI）		円周率を指定（π≒3.141592654）

2．BASIC命令

（※……プログラムで実行可能な命令
★……マニュアル操作で実行可能な命令）

	命　令	省略形	一　般　形	備　考
※	AREAD (オートリード)	A. 〜 AREA.	AREAD　変数	定義付けキーによるプログラム実行開始時に、表示内容を指定された変数に入れる。
※ ★	BEEP (ビープ)	B. 〜 BEE.	BEEP　式	音発生命令
※ ★	CLEAR (クリア)	CL. 〜 CLEA.	CLEAR	データ（変数）をすべて消去する命令。

	命　令	省略形	一　般　形	備　考
※	DATA (データ)	DA. DAT.	DATA　式，式，…	データ指定 READ命令に続く変数に与えるデータを指定。
※ ★	DEGREE (ディグリー)	DE. 〜 DEGRE.	DEGREE	角度単位指定 "度"単位指定〔°〕
※ ★	DIM (ディメンジョン)	D. DI.	(1)DIM 変数名(式) (2)DIM 変数名(式)＊式$_3$ (3)DIM 変数名(式$_1$，式$_2$) (4)DIM 変数名(式$_1$，式$_2$)＊式$_3$ 変数名：B、C$、D$など (　　)：配列の大きさ、次元指定 式$_3$　：桁数指定(文字変数のみ)	変数の定義 配列変数の変数名、大きさなどを定義し、プログラム・データエリアに確保する。
※	END (エンド)	E. EN.	END	プログラム終了命令
※	FOR (フォー) TO (トゥー) STEP (ステップ)	F. FO. STE.	(1)FOR　数値変数＝式$_1$　TO　式$_2$ (2)FOR 数値変数＝式$_1$ TO 式$_2$ STEP 式$_3$ 式$_1$：初期値，式$_2$：最終値 式$_3$：増（減）分	FOR－NEXTループの始まり NEXT命令と対で使用
※	GOSUB (ゴーサブ)	GOS. GOSU.	GOSUB { 式 / "文字" / 文字変数 } ●指定されたラインあるいはラベルへ実行を移し、そこからのプログラムをサブルーチンとして実行。 RETURN命令と対で使用。	サブルーチンジャンプ命令
※ ★	GOTO (ゴートゥ)	G. 〜 GOT.	GOTO { 式 / "文字" / 文字変数 } ●指定されたラインあるいはラベルへ実行を移す。 ●指定されたラインからプログラムを実行する。	ジャンプ命令 （プログラム実行開始命令）

	命　令	省略形	一　般　形	備　考
※ ★	GRAD (グラード)	GR. GRA.	GRAD	角度単位指定 "グラード"単位指定 〔g〕
※	IF (イフ)		IF { 条件式 / 式 } 実行文	判断命令 与えられた条件を判断し、実行を次の実行文か、あるいは次のラインへ移します。
※	INPUT (インプット)	I. 〜 INPU.	(1) INPUT変数,変数,… (2) INPUT"文字",変数,"文字",変数,… (3) INPUT"文字";変数,"文字";変数,…	入力命令 データを入力
※	LET (レット)	LE.	(1)LET数値変数=式 (2)LET文字変数="文字" (3)LET文字変数=文字変数	LETはIF文に続く場合以外は省略可
※ ★	RADIAN (ラディアン)	RAD. 〜 RADIA.	RADIAN	角度単位指定 "ラディアン"単位指定〔rad〕
※ ★	RANDOM (ランダマイズ)	RA. 〜 RANDO.	RANDOM	RND命令に先だって乱数のタネを植えつけるものです。
※	READ (リード)	REA.	READ変数, 変数, …	データの読み込み命令 DATA命令に続いて指定されているデータを指定の変数に入れる。
※	REM (リマーク)		REM注釈	プログラムの中に非実行文（注釈）を指定するための命令。
※	RESTORE (リストア)	RES. 〜 RESTOR.	(1)RESTORE { 式 / "文字" / 文字変数 } (2)RESTORE	READ命令により読み込まれるデータの順番を変える命令。

	命　令	省略形	一　般　形	備　考
※	RETURN (リターン)	RE. 〜 RETUR.	RETURN	復帰命令 サブルーチンの復帰命令で、GOSUBの次のステートメントへ実行が復帰します。
※	STOP (ストップ)	S. 〜 STO.	STOP	プログラムの実行を停止する命令。
※	THEN (ゼン)	T. 〜 THE.	IF文における実行文として定義します。 THEN { 式 / "文字" / 文字変数 } THEN{BASIC命令}	IF文における実行文として定義することのみ可能。
※	LPRINT (ラインプリント)	LP. 〜 LPRIN.	(1)LPRINT { 式 / "文字" / 文字変数 } (2)LPRINT { 式 / "文字" / 文字変数 }, { 式 / "文字" / 文字変数 } (3)LPRINT { 式 / "文字" / 文字変数 }: { 式 / "文字" / 文字変数 }…	出力命令 指定された内容を印字します。
※	NEXT (ネクスト)	N. 〜 NEX.	NEXT　数値変数 数値変数はFOR命令に続く数値変数と一致していなければならない。	FOR－NEXTループの末端を示します。FOR文と対で使用。
※	ON GOSUB (オンゴーサブ)	ONGOS. O. GOS. O. GOSU.	ON　式　GOSUB式$_1$, 式$_2$, 式$_3$,…	サブルーチンジャンプ命令。 ONに続く式の値によりジャンプ先を指定。
※	ON GOTO (オンゴートゥー)	ONG. O. G. 〜 O. GOT.	ON　式　GOTO式$_1$, 式$_2$, 式$_3$,…	シャンプ命令 ONに続く式の値によりジャンプ先を指定する。
※	PAUSE (ポーズ)	PAU. PAUS.	一般形はPRINT命令と同じ形をとります	出力命令 指定された内容を約0.85秒表示したのちプログラムを実行。

	命　令	省略形	一　般　形	備　考
⁘	PRINT (プリント)	P. 〜 PRIN.	(1)PRINT { 式 / "文字" / 文字変数 } (2)PRINT { 式 / "文字" / 文字変数 } , { 式 / "文字" / 文字変数 } (3)PRINT { 式 / "文字" / 文字変数 } ; { 式 / "文字" / 文字変数 } …	出力命令 指定された内容を表示します。
⁘ ★			PRINT=LPRINT PRINT=PRINT	PRINT命令をLPRINT命令に指定 PRINT→LPRINTを解除
※ ★	TR OFF (トレースオフ)	TROF.	TR OFF	デバッグをおこなうモードを解除
※ ★	TR ON (トレースオン)	TR. TRO.	TR ON	デバッグをおこなうモードを指定。
※ ★	USING (ユージング)	U. 〜 USIN.	(1)USING"#…#. #…#^" (2)USING"&&&……&&&" (3)PRINT USING"フォーマット";… (4)USING (5)PRINT USING;…	フォーマット指定命令 数値データ、文字データの出力フォーマット指定 フォーマット指定解除
※ ★	WAIT (ウェイト)	W. 〜 WAI.	(1)WAIT 式 (式の値は 0〜65535) (2)WAIT	表示時間指定 PRINT命令によるプログラム停止時間を指定。 指定を解除。 (停止時間無限)

3．コマンド(マニュアル操作でのみ実行可能)

	命　令	省略形	一　般　形	備　考
★	CONT (コンティニュー)	C. 〜 CON.	CONT	一時停止しているプログラムの実行を再開する命令 RUNモードで有効。
★	LIST (リスト)	L. 〜 LIS.	(1)LIST (2)LIST{式 / "文字" / 文字変数}	プログラムリストをおこなう命令。 PROモードで有効
★	LLIST (ラインリスト)	LL. 〜 LLIS.	(1)LLIST (2)LLIST式$_1$, 式$_2$	リスト命令 プリンタによりプログラムのリストアウトをおこなう命令。
★	NEW (ニュー)		NEW	PROモードでプログラムおよび全変数をクリア。
★	PASS (パス)	PA. PAS.	PASS　"文字"	パスワードの設定、解除をする命令。
★	RUN (ラン)	R. RU.	(1)RUN (2)RUN式 (3)RUN{"文字" / 文字変数}	プログラム実行開始命令。 RUNモードで有効

4．BASIC関数・その他

	命　令	省略形	一　般　形	備　考
※ ★	AND (アンド)	AN.	式　AND　式	論理積
※ ★	ASC (アスキー)		ASC{"文字" / 文字変数}	文字などのキャラクタをアスキーコードに変換。

	命　令	省略形	一　般　形	備　考
※★	CHR$（キャラクタドル）	CHR.	CHR$ 式	アスキーコードを文字などのキャラクタに変換。
※★	INKEY$（インキードル）	INK. 〜 INKEY.	文字変数＝INKEY$	INKEY$命令実行時にキーが押されていればその内容を読み込む。
※★	LEFT$（レフトドル）	LEF. LEFT.	LEFT$（文字変数，式） LEFT$（“文字”，式）	指定された文字列の左側から、指定された桁数の文字を取り出す
※★	LEN（レングス）		LEN{“文字” / 文字変数}	指定された文字列の文字数を求めます。
※★	MEM（メム）	M. ME.	MEM	プログラム・データエリアの中で空いている部分のバイト数を表示。
※★	MID$（ミッドドル）	MI. MID.	MID$（文字変数，式$_1$，式$_2$） MID$（“文字”，式$_1$，式$_2$）	指定された文字列の中間の文字を取り出す。
※★	NOT（ノット）	NO.	NOT 式	否　定
※★	OR（オア）		式 OR 式	論理和
※★	RIGHT$（ライトドル）	RI. 〜 RIGHT.	RIGHT$（文字変数，式） RIGHT$（“文字”，式）	指定された文字列の右側から、指定された桁数の文字を取り出す。
※★	RND（ランダム）	RN.	RND 式	乱数を得る命令。
※★	STR$（ストリングドル）	STR.	STR$ 式 （A$＝STR$ A）	数値を文字列に変換
※★	VAL（バリュー）	V. VA.	VAL{“文字” / 文字変数}	文字列を数値に変換

5．テープレコーダーに関する命令

	命　令	省略形	一　般　形	備　考
※	CHAIN （チェイン）	CH. 〜 CHAI.	(1)CHAIN “ファイル名” (2)CHAIN “ファイル名”，{式 / “ラベル”} (3)上記(1),(2)の“ファイル名”，を省略した形 （CHAINに続く“ラベル”はファイル名と見なされます。）	転送命令 プログラムの中で使用する命令。プログラムをテープから読み込んで（転送して）そのプログラムを実行。
★	CLOAD （カセットロード）	CLO. CLOA.	(1)CLOAD (2)CLOAD “ファイル名”	転送命令 プログラムをテープから計算機に転送する命令。
★	CLOAD? （カセットロード?）	CLO. ? CLOA. ?	(1)CLOAD? (2)CLOAD? “ファイル名”	照合命令 計算機内のプログラムとテープに記録されている内容との照合をおこなう命令。
※ ★	CSAVE （カセットセイブ）	CS. 〜 CSAV.	(1)CSAVE (2)CSAVE “ファイル名” (3)CSAVE“ファイル名”,“パスワード” (4)CSAVE,“パスワード”	記録命令 プログラムをテープに記録する命令。 {プログラムを秘密化する。
※ ★	INPUT＃ （インプット クロスハッチ）	I. ＃ 〜 INPU. ＃	(1) INPUT＃“ファイル名” (2) INPUT＃“ファイル名”; 変数名， 変数名，… (3)上記(1),(2)のファイル名を省略した形	データ転送命令 テープに記録されているデータを指定された変数に転送する命令。
★	MERGE （マージ）	MER. MERG.	一般形はCLOAD命令と同じ形となります。 （この命令では、それまで記憶していたプログラムはそのまま保持し、その後に新しく読み込んだプログラムが記憶される。）	転送命令 プログラムをテープから計算機に転送する命令。
※ ★	PRINT＃ （プリント クロスハッチ）	P. ＃ 〜 PRIN. ＃	一般形はINPUT＃命令と同じ形となります。	データ記録命令 計算機に記憶されているデータをテープに記録する命令。

索　引

（ページ）

ABS……16
ACS……16
AND……60、62
AREAD……56
ASC……69
ASN……16
ATN……16
BEEP……59
CHAIN……80
CHR$……69
CLEAR……46
CLOAD……77
CLOAD?……81
CONT……59
COS……16
CSAVE……76
DATA……53
DEG……16
DEGREE……16
DEL……14
DIM……42
DMS……16
END……34
EXP……16
FOR……37
GOSUB……39
GOTO……34
GRAD……16
IF……35
INKEY$……58
INPUT……30
INPUT#……83
INS……15
INT……16

（ページ）

LEFT$……66
LEN……67
LET……33
LIST……28
LLIST……74
LN……16
LOG……16
LPRINT……72
MEM……60
MERGE……78
MID$……67
NEW……25
NEXT……37
NOT……61
NUL……46
ON GOSUB……55
ON GOTO……54
OR……61、63
PASS……71
PAUSE……51
PI……16
PRINT……32
PRINT#……82
RADIAN……16
RANDOM……58
READ……53
REM……55
RESTORE……54
RETURN……39
RIGHT$……66
RND……57
RUN……28
SGN……16
SIN……16

（ページ）

SQR…… 16
STEP…… 37
STOP…… 55
STR$…… 68
TAN…… 16
THEN…… 35
TO…… 37
TR OFF…… 64
TR ON…… 64
USING…… 49
VAL…… 68
WAIT…… 51

アスキーコード…… 69
アブソリュート…… 16
一次元配列…… 42
インサート…… 15
インテジャー…… 16
エラーコード…… 133
エラー表示…… 13、29
オートパワーオフ…… 6
音発生機能…… 59
カーソル…… 8、14
仮数部…… 13
キャラクタ…… 69、135
きざみ…… 37
グラード…… 16
固定変数…… 41
最終値…… 37
サブルーチン…… 39
初期値…… 37
シグナム…… 16
指数方式…… 13
指数部…… 13
ジャンプ…… 34
条件式…… 35
スキップ…… 32

（ページ）

ステートメント…… 25
定義付け…… 48
データ専用エリア…… 44
データバッファ…… 22
データメモリー…… 19
デバッグ…… 64
デリート…… 14
テンプレート…… 7
トレースモード…… 64
度…… 16
二次元配列…… 42
配列変数…… 42
判断…… 35
バッファ…… 22
バイト…… 11
パスワード…… 71
表示フォーマット…… 49
否定…… 61
秘密プログラム…… 71
ファイル名…… 76
ファンクションバッファ…… 22
浮動小数点方式…… 13
プレイバック…… 15
プログラム・データエリア…… 44
プログラムモード…… 9
プロンプト記号…… 8
変数…… 19、41
メインルーチン…… 39
文字配列変数…… 42
ライン…… 24
ラディアン…… 16
ラベル…… 48
ランモード…… 9
論理演算子…… 19
論理積…… 60
論理和…… 61

＊ ＊ メ モ ＊ ＊

保　証　書（保証規定）

本書は、本書記載内容で無料修理をさせて頂くことをお約束するものです。
保証期間中に故障が発生した場合は、製品と本書をご持参、ご提示の上、お買上げの販売店にご依頼ください。
お買上げ年月日、販売店名など記入もれがありますと無効となります。必ずご確認いただき、記入のない場合はお買上げの販売店にお申し出ください。
ご転居・ご贈答品などでお買上げ販売店に修理をご依頼できない場合は、製品に同梱しております「シャープサービス・お客様ご相談窓口一覧表」をご覧の上、もよりのサービス会社へご持参ください。
本書は再発行いたしません。たいせつに保管してください。

〈無料修理規定〉

1．取扱説明書・本体注意ラベルなどの注意書にしたがった正常な使用状態で、保証期間内に故障した場合には、お買上げ販売店、または当社サービス会社が無料修理いたします。

2．保証期間内でも次の場合には有料修理となります。
（イ）本書のご提示がない場合。
（ロ）本書にお買上げ年月日・お客様名・販売店名の記入がない場合、または字句を書き換えられた場合。
（ハ）使用上の誤り、または不当な修理や改造による故障・損傷。
（ニ）お買上げ後に落とされた場合などによる故障・損傷。
（ホ）火災・公害および地震・風水害その他天災地変など、外部に要因がある故障・損傷。
（ヘ）電池の液もれによる故障・損傷。
（ト）消耗品(リチウム電池)が損耗し取り替えを要する場合。

3．本書は日本国内においてのみ有効です。
（THIS WARRANTY CARD IS ONLY VALID FOR SERVICE IN JAPAN.）

★この保証書は本書に明示した期間・条件のもとにおいて無料修理をお約束するものです。したがいましてこの保証書によってお客様の法律上の権利を制限するものではありませんので、保証期間経過後の修理などにつきましておわかりにならない場合はお買上げの販売店、またはシャープお客様ご相談窓口にお問い合せください。

修理メモ

シャープ計算機保証書
(WARRANTY CARD)

品　　名　計　算　機

形　　名　PC-1245

保証期間 (VALIDITY)　お買上げ日より本体1年間 (FULL 1 YEAR AFTER PURCHASE)

お買上げ日 (PURCHASE)　昭和　年　月　日

検査日 59. 6. 9

お客様		
	ご芳名	様
	ご住所	〒
	電話番号	(　　)　　—

取扱販売店名・住所・電話番号

家電・エレクトロニクスの

ダイイチ本店

事務機係

広島市中区紙屋町2-1-18　TEL 247—5111

本社 広島市中区紙屋町2-1-18 第一産業㈱

印

シャープ株式会社

産業機器事業本部

〒639-11　奈良県大和郡山市美濃庄町492

電話 (07435)3—5521番

シャープ株式会社

本　　社　〒545 大阪市阿倍野区長池町22－22
　　電話（06）621－1221（大代表）
産業機器事業本部　〒639-11 奈良県大和郡山市美濃庄町492
　　電話（07435）3－5521（大代表）
国内産機営業本部　〒162 東京都新宿区市谷八幡町８番地
　　電話（03）260－1161（大代表）

プログラムや製品、機能についてのお問い合わせは下記へ……

シャープビジネス株式会社

〒545 大阪市阿倍野区長池町22－22　電話（06）621－1221（大代表）

札幌支店	〒063	札幌市西区二十四軒1条7丁目3番17号	電話（011）641－3631
仙台支店	〒983	仙台市6丁目字本新田東2番の1	電話（0222）88－9151
宇都宮支店	〒320	宇都宮市不動前4丁目2番41号	電話（0286）37－3576
東京支店	〒130	東京都墨田区石原2丁目12番3号	電話（03）625－5111
千葉支店	〒280	千葉市南町1丁目5番20号	電話（0472）63－4043
横浜支店	〒235	横浜市磯子区中原1丁目2番23号	電話（045）751－3215
埼玉支店	〒330	埼玉県大宮市宮原町2丁目107番地2号	電話（0486）63－5159
新潟支店	〒950	新潟市上所中1丁目7番21号	電話（0252）83－1795
長野支店	〒380	長野市中御所2丁目11番3号	電話（0262）28－4618
名古屋支店	〒454	名古屋市中川区山王3丁目5番5号	電話（052）332－2631
金沢支店	〒921	石川県石川郡野々市町御経塚1096番1号	電話（0762）49－1240
大阪支店	〒556	大阪市浪速区恵美須西1丁目2番9号	電話（06）643－3021（代表）
京都支店	〒601	京都市南区上鳥羽管田町48番地	電話（075）661－7335
神戸支店	〒658	神戸市東灘区魚崎北町1丁目6番地18号	電話（078）452－8531
広島支店	〒731-01	広島市安佐南区祇園町大字西原2249番地の1	電話（082）874－4925
高松支店	〒760	香川県高松市木太町1861番地の3	電話（0878）33－4255
福岡支店	〒816	福岡市博多区井相田2丁目12番地の1	電話（092）572－2611
沖縄支店	〒900	沖縄県那覇市曙2丁目10番地の1	電話（0988）61－7360（代表）

修理などアフターサービスについてのお問い合わせは下記へ……

シャープお客様ご相談窓口（技術センター）

札幌	技術センター	〒（063） 札幌市西区二十四軒1条7丁目3番17号	電話（011）641－0751
仙台	技術センター	〒（983） 仙台市6丁目字本新田東2番の1	電話（0222）88－9161
宇都宮	技術センター	〒（320） 宇都宮市不動前4丁目2番41号	電話（0286）34－0256
新潟	技術センター	〒（950） 新潟市上所中1丁目7番21号	電話（0252）84－6023
東京第１	技術センター	〒（130） 東京都墨田区石原2丁目12番3号	電話（03）624－7475
東京第２	技術センター	〒（164） 東京都中野区南台3丁目45番13号	電話（03）382－9161
東京第３	技術センター	〒（143） 東京都大田区南馬込1丁目5番15号	電話（03）777－8851
横浜	技術センター	〒（235） 横浜市磯子区中原1丁目2番23号	電話（045）753－9583
名古屋	技術センター	〒（454） 名古屋市中川区山王3丁目5番5号	電話（052）332－2671
金沢	技術センター	〒（921） 石川県石川郡野々市町字御経塚町1096の1	電話（0762）49－9033
大阪第１	技術センター	〒（556） 大阪市浪速区恵美須西1丁目2番9号	電話（06）644－1191
大阪第２	技術センター	〒（567） 茨木市鮎川5丁目15番3号	電話（0726）34－4683
神戸	技術センター	〒（658） 神戸市東灘区魚崎北町1丁目6番18号	電話（078）452－1762
京都	技術センター	〒（601） 京都市南区上鳥羽管田町48番地	電話（075）681－9551
高松	技術センター	〒（761-01） 高松市高松町3009番地4	電話（0878）43－3711
岡山	技術センター	〒（700） 岡山市米倉字東ノ町66の2	電話（0862）43－5660
広島	技術センター	〒（731-01） 広島市安佐南区祇園町大字西原2249番地の1	電話（082）874－6100
福岡	技術センター	〒（816） 福岡市博多区井相田2丁目12番地1号	電話（092）572－2617

©1983年

4D18.0O〈TINSJ3904CCZZ〉⑨

SHARP

全国をくまなくめぐるシャープのサービス網

このたびはシャープ製品をお買上げいただきましてありがとうございます。
シャープ製品のアフターサービスはお買上げ販売店、または下記シャープシステムサービス㈱がさせていただきます。
なお、アフターサービスについておわかりにならないことは、もよりの技術センターへお問合せください。その他ご相談、お問合せは各地のお客様ご相談窓口をご利用ください。
この一覧表は取扱説明書、保証書などと共に保存してください。

シャープサービス・お客様ご相談窓口一覧表

都道府県名	サービス拠点名	電話番号	郵便番号	所在地
北海道	**札幌技術センター**	(011)641−0751	063	札幌市西区二十四軒1条7丁目3番17号
	札幌技術センター 帯広	(0155)35−4068	080	帯広市西11条南1丁目12番地1
	札幌技術センター 旭川	(0166)22−8284	070	旭川市一条通4丁目左10号
	札幌技術センター 函館	(0138)52−5190	040	函館市五稜郭町31番17号
	札幌技術センター 釧路	(0154)24−8191	085	釧路市光陽町8番地の13
	札幌技術センター 北見	(0157)36−6814	090	北見市三輪町435番地
	札幌技術センター 苫小牧	(0144)74−8070	053	苫小牧市有明町1丁目5番6号
宮城	**仙台技術センター**	(0222)88−9161	983	仙台市六丁目字本新田東2番地の1
福島	仙台技術センター 郡山	(0249)46−0196	969−05	郡山市安積町荒井字方八丁33番地の1
	仙台技術センター いわき	(0246)28−2487	970	いわき市自由ケ丘37番地の10
岩手	仙台技術センター 盛岡	(0196)38−6805	028−36	岩手県紫波郡矢巾町流通センター南3丁目1番1号
	仙台技術センター 水沢	(0197)24−1710	023	水沢市羽町3番10号
秋田	仙台技術センター 秋田	(0188)65−1258	010	秋田市川尻町字大川反170番地の56
青森	仙台技術センター 青森	(0177)38−7778	030	青森市安方3丁目3番4号
	仙台技術センター 八戸	(0178)45−2631	031	八戸市小中野2丁目8番16号
山形	仙台技術センター 山形	(0236)31−6634	990−23	山形市蔵王飯田227番地1
	シャープ事務機山形販売㈱	(0236)33−3215	990−22	山形市漆通センター1丁目9番地の2
栃木	**宇都宮技術センター**	(0286)34−0256	320	宇都宮市不動前4丁目2番41号
	宇都宮技術センター 小山	(0285)25−6532	323	小山市城東1丁目17番10号
群馬	宇都宮技術センター 前橋	(0272)52−7311	371	前橋市問屋町1丁目3番7
	宇都宮技術センター 太田	(0276)46−4110	373	太田市東長岡1788
茨城	宇都宮技術センター 水戸	(0292)43−0909	310	水戸市千波町1963番地
	宇都宮技術センター 土浦	(0298)23−7673	300	土浦市真鍋町6丁目20祖2255番地
新潟	**新潟技術センター**	(0252)84−6023	950	新潟市上所中1丁目7番21号
	新潟技術センター 長岡	(0258)23−1850	940−11	長岡市摂田屋町字狼2500番
	新潟技術センター 上越	(0255)22−3360	943	上越市大字土橋字大坪1905番地の1
長野	新潟技術センター 松本	(0263)27−1636−7	399−65	松本市芳野8番14号
	新潟技術センター 長野	(0262)26−6208	380	長野市中御所2丁目11番3号
	新潟技術センター 上田	(0268)27−9395	386	上田市住吉67−16
東京	第1技術部・第2技術部	(03) 260−1161(大代表)	162	新宿区市谷八幡町8番地シャープ東京ビル内
	東京第1技術センター	(03) 624−7475	130	墨田区石原2丁目12番3号シャープ江東ビル内
	東京第1技術センター 城東	(03) 606−4330	120	足立区綾瀬7丁目22番11号
	東京第2技術センター	(03) 382−9161	164	中野区南台3丁目45番13号 シャープ城西ビル内
	東京第2技術センター 城北	(03) 973−5120	174	板橋区東新町1丁目33番11号
	東京第2技術センター 武蔵野	(03) 326−2181	157	世田谷区給田町5丁目19−1
	東京第2技術センター 日野	(0425)85−1656	191	日野市日野台5丁目5番4号
	東京第3技術センター	(03) 777−8851	143	大田区南馬込1丁目5番15号シャープ城南ビル内
埼玉	**大宮技術センター**	(0486)66−7220	330	大宮市宮原町2丁目107番2号
	大宮技術センター 熊谷	(0485)54−5140	361	行田市門井町2丁目5番地
	大宮技術センター 戸田	(0484)43−2788	335	戸田市上戸田3丁目26番6号
千葉	東京第1技術センター 千葉	(0472)65−6110	280	千葉市南町1丁目5番20号
	東京第1技術センター 松戸	(0473)68−5166	271	松戸市稔台295番1
	東京第1技術センター 船橋	(0474)23−7170	273	船橋市市場4丁目16番2号
山梨	東京第2技術センター 甲府	(0552)28−1320	400	甲府市富竹2丁目1番17号
神奈川	**横浜技術センター**	(045)753−9583	235	横浜市磯子区中原1丁目2番23号
	横浜技術センター 湘南	(0463)55−7600	254	平塚市田村字富向1381
	横浜技術センター 相模原	(0427)59−4398	229	相模原市淵野辺3−1−11
静岡	横浜技術センター 静岡	(054)283−9497	422	静岡市曲金6丁目8番44号
	横浜技術センター 沼津	(0559)24−1028	410	沼津市宮前町11−4
	名古屋技術センター 浜松	(0534)65−0735	430	浜松市植松町1476−2
愛知	**名古屋技術センター**	(052)332−2671	454	名古屋市中川区山王3丁目5番5号シャープ名古屋ビル内
	名古屋技術センター 豊橋	(0532)54−1830	440	豊橋市下地町橋口170−1
	名古屋技術センター 半田	(0569)21−1369	475	半田市住吉町2−109
	名古屋技術センター 岡崎	(0564)25−0611	444	岡崎市柿田町1番地21
岐阜	名古屋技術センター 岐阜	(0582)74−7996	500	岐阜市六条南3丁目12番9号
	名古屋技術センター 濃飛	(0574)25−8418	509−02	可児市土田字下切3832−1
三重	名古屋技術センター 津	(0592)31−1573	514−01	津市栗真町屋町字蒲池328番地
	名古屋技術センター 四日市	(0593)51−9434	510	四日市市鵜の森1丁目15番地9
石川	**金沢技術センター**	(0762)49−9033	921	石川県石川郡野々市町字御経塚町105の1
富山	金沢技術センター 富山	(0764)51−3933	930	富山市金泉寺71番1号
福井	金沢技術センター 福井	(0776)53−6050	910	福井市北四ツ居町625番地
	シャープ事務機福井販売㈱	(0776)27−1800	910	福井市新田塚1丁目70番26号
大阪	第4技術部・第5技術部	(06) 621−1221(大代表)	545	大阪市阿倍野区長池町22番22号
	大阪第1技術センター	(06) 644−1191	556	大阪市浪速区恵美須西1丁目2番9号
	大阪第1技術センター 東大阪	(0729)93−2001	581	八尾市跡部北の町3丁目2番11号
	大阪第1技術センター 南大阪	(0722)45−5855	590	堺市老松町1丁目39番地
	大阪第2技術センター	(0726)34−4683	567	茨木市鮎川5丁目15番3号
奈良	大阪第1技術センター 奈良	(07435)3−2023〜4	639−11	大和郡山市美濃庄町492番地
和歌山	大阪第1技術センター 和歌山	(0734)45−6298	641	和歌山市西小二里2丁目4番91号
	大阪第1技術センター 田辺	(0739)24−4092	646	田辺市稲成町441−1
兵庫	**神戸技術センター**	(078)452−1762	658	神戸市東灘区魚崎北町1丁目6番18号
	神戸技術センター 姫路	(0792)66−8295	671−22	姫路市青山字道之内1006番の1
	神戸技術センター 阪神	(06) 421−2304	661	尼崎市猪名寺西田187番地
	神戸技術センター 豊岡	(07962)2−6174	668	豊岡市九日市上町下垣77の1
京都	**京都技術センター**	(075)681−9551	601	京都市南区上鳥羽雲田町48番地
	京都技術センター 舞鶴	(0773)76−6069	624	舞鶴市福来間屋町785−47
滋賀	京都技術センター 大津	(0775)43−2331	520−21	大津市栗林町11−35
	京都技術センター 彦根	(0749)23−8688	522	彦根市古沢町406番地の1
香川	**高松技術センター**	(0878)43−3711	761−01	高松市高松町3009番地4
愛媛	高松技術センター 松山	(0899)73−0121	791−41	松山市恵岡町178番1
	高松技術センター 新居浜	(0897)43−8854	792	新居浜市国領1丁目甲1799番地の4
	高松技術センター 宇和島	(0895)24−1975	798	宇和島市中沢町1丁目1番20号
徳島	高松技術センター 徳島	(0886)25−6840	770	徳島市中常三島町3丁目11番地14
高知	高松技術センター 高知	(0888)83−7039	780	高知市高須字高須塩田西の丸960番1
岡山	**岡山技術センター**	(0862)43−5660	700	岡山市米倉字東ノ町686の2
	岡山技術センター 津山	(08682)2−6296	708	津山市沼6番地の8
鳥取	岡山技術センター 鳥取	(0857)26−4227	680	鳥取市青葉町2丁目204番地
広島	岡山技術センター 福山	(0849)52−0736	720	福山市津之郷町大字津之郷字上開地
	広島技術センター	(082)874−6100	731−01	広島市安佐南区西原2丁目13番4号
	広島技術センター 東広島	(0824)28−3061	739−01	東広島市八本松町大字米満字徳政35−3
山口	広島技術センター 徳山	(0834)32−3839	745	徳山市新地1番地
	広島技術センター 下関	(0832)56−6699	751	下関市秋根南町2丁目5番2号
	広島技術センター 小郡	(08397)2−4525	754	吉敷郡小郡町大字下郷2255−39
島根	広島技術センター 松江	(0852)21−6110	690	松江市西津田3丁目1番10号
福岡	**福岡技術センター**	(092)572−2617	816	福岡市博多区井相田2丁目12番地1号
	福岡技術センター 北九州	(093)592−6510	803	北九州市小倉北区大手町6番12号
	福岡技術センター 久留米	(0942)21−7858	830	久留米市上津町1682−2
	福岡技術センター 筑豊	(0948)24−9283	820	飯塚市大字川津字宮の前272−2
大分	福岡技術センター 大分	(0975)32−0518	870	大分市中春日町16番20号
長崎	福岡技術センター 長崎	(0958)45−8481	852	長崎市宝町6番11号
	福岡技術センター 佐世保	(0956)33−7288	857−11	佐世保市白岳町107−5
佐賀	福岡技術センター 佐賀	(0952)25−0983	840	佐賀市鍋島町大字八戸字五本松篭2043−2
鹿児島	**鹿児島技術センター**	(0992)59−0628	890	鹿児島市鴨池新町12番1号
宮崎	鹿児島技術センター 宮崎	(0985)28−8371	880	宮崎市原町4番12号
	鹿児島技術センター 延岡	(0982)21−7744	882	延岡市浜砂町11番番地
熊本	鹿児島技術センター 熊本	(0963)72−1251	862	熊本市新屋敷3丁目15番17号
沖縄	福岡技術センター 沖縄	(0988)66−5562	900	那覇市曙2丁目10番地の1

※ 上記の所在地、電話番号は変わることがありますので、その節はご容赦ください